

Faculdade de Engenharia da Universidade do Porto



**Bancada on-line para o ensino de
Microprocessadores**

Vitor Hugo Fernandes Torres

VERSÃO PROVISÓRIA

Dissertação realizada no âmbito do
Mestrado Integrado em Engenharia Electrotécnica e de Computadores
Major Automação

Orientador: Prof. Dr. José Manuel Martins Ferreira

Julho de 2008

A Dissertação intitulada

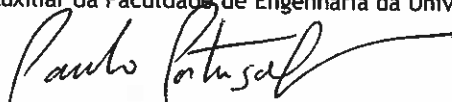
“Bancada on-line para o ensino de microprocessadores”

foi aprovada em provas realizadas em 17/Julho/2008

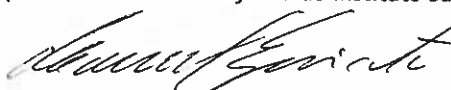
o júri

Presidente

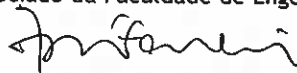
Professor Doutor Paulo José Lopes Machado Portugal
Professor Auxiliar da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Manuel Gradim de Oliveira Gericota
Equiparado a Professor Adjunto do Instituto Superior de Engenharia do Porto

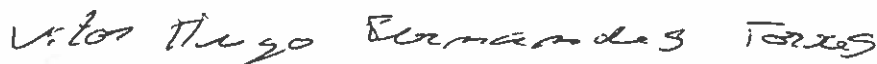


Professor Doutor José Manuel Martins Ferreira
Professor Associado da Faculdade de Engenharia da Universidade do Porto



O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados.-

Autor - Vitor Hugo Fernandes Torres



Faculdade de Engenharia da Universidade do Porto

Resumo

O objectivo deste trabalho é desenvolver uma solução que permita aos alunos da disciplina de Microprocessadores (EEC0029) realizar trabalhos práticos, a qualquer hora e a partir de qualquer local. A solução pretendida é apoiada em cartas já existentes, baseadas no microcontrolador 80C51, e recorre a interfaces Web desenvolvidas em *LabView* ou em outra tecnologia apropriada para este efeito. Toma-se como ponto de partida um protótipo já existente e apresenta-se uma solução que permite a realização de todas as experiências previstas na disciplina de Microprocessadores, com prévia reserva do acesso à bancada.

A tecnologia de desenvolvimento de bancadas on-line utilizada foi o conjunto *LabView + NI ELVIS* da *National Instruments*. No desenvolvimento da bancada para a família 80C51, foi ainda utilizada a carta *CORE51*, que é frequentemente utilizada na disciplina de Microprocessadores.

Foram desenvolvidos em *LabView* três painéis frontais que permitem cobrir a totalidade das experiências realizadas na cadeira de Microprocessadores. O acesso a estes painéis foi feito via Web através da ferramenta *Web Publishing Tool*, integrada no *LabView* e que permite o acesso remoto aos painéis que se encontram em execução no servidor de laboratório.

O VI de comunicação com a *CORE51* possibilita a transferência do código em formato Intel HEX, bem como o relatório sobre o correcto funcionamento da carta, consoante o modo de operação escolhido.

Quanto ao VI de interacção com a carta, para além da escolha dos modos de funcionamento da *CORE51*, permite ainda o controlo de seis interruptores, bem como a publicação em tempo real de uma imagem da experiência, através de uma câmara Web.

Por último, o VI de instrumentação disponibiliza a utilização de um osciloscópio de dois canais aliado a um gerador de funções simplificado, instrumentos estes que são de grande utilidade na maioria das disciplinas do MIEEC.

Quanto ao mecanismo de reserva da bancada, foi utilizado o bloco *Metting Room Booking System* (MRBS) e foram efectuadas alterações no código a fim de esta ferramenta ir de encontro com os requisitos impostos. São exemplos dessas alterações o lançamento da hiperligação de acesso aos VI's, bem como a modificação da interface base do bloco. Esta ferramenta foi integrada no servidor de *e-learning* de testes da FEUP (<http://moodle.fe.up.pt/dev0809>), de modo a permitir a autenticação via *LDAP* dos utilizadores.

Abstract

The objective of this work is to develop an on-line microcontroller / microprocessor workbench that enables the students to do their lab assignments at any time and from anywhere. The remote experiment hardware must reuse already existing 80C51 boards and the user interfaces should be based on LabView or other equivalent technology. An existing prototype was adopted as the initial solution and was customized to meet the functional requirements presented in this document. All standard lab assignments traditionally used in the EEC0029 Microprocessors course are supported by this on-line workbench.

The solution presented in this document is based on National Instruments' LabView programming language and ELVIS lab stations. The current version of the remote experiment hardware uses the CORE51 microcontroller board, which has been used several times in the EEC0029 course.

The three Virtual Instrument (VI) interface panels that were developed enable all the lab assignments that are traditionally proposed to the students within the EEC0029 course. The LabView Web Publishing Tool was used to enable access via the web to these VIs that are executed on the campus lab server.

The communications VI enables code transfer in Intel Hex format and displays status information that indicates the current operating mode of the CORE51 board.

The interaction VI enables the user to select the required CORE51 operating mode and to control six digital inputs connected to microcontroller I/O lines. It also displays a real-time image of the remote experiment hardware, captured with a webcam located in the on-line workbench.

Finally the instrumentation VI offers a two-channel oscilloscope and a simplified waveform generator. These two tools are frequently used beyond the typical microcontroller lab assignments and make this on-line workbench useful in a wider range of introductory digital and analogue electronics courses.

The Meeting Room Booking System (MRBS) was adopted as a basis for the development of a Moodle-embedded scheduling solution that enables the students to book access time to the on-line workbenches.

Agradecimentos

Com os melhores agradecimentos para o meu professor orientador, José Manuel Martins Ferreira, por todo o apoio, ajuda e conselhos a mim prestados durante todo o percurso de desenvolvimento da dissertação.

Quero agradecer com especial carinho à minha família, que em tudo me ajudou e à qual devo a possibilidade de ter ingressado no ensino superior, pois sem ela tudo o que fiz e o tudo o que sou não seria possível. Devo particular agradecimento ao meu pai, que se sacrificou em trabalhar no estrangeiro a fim de me proporcionar todas as condições para que cumprisse o curso com sucesso.

Agradeço com um forte abraço e o desejo de um futuro repleto de sucesso a todos os meus amigos, que me deram o apoio e motivação necessários neste percurso que nem sempre nos proporciona somente alegrias.

E por fim mas não menos importante, um especial agradecimento à Filipa Moure, que acompanhou uma grande parte do meu trajecto académico e cujo apoio foi irredutível a tempo inteiro.

Índice

Resumo.....	iii
Abstract	v
Agradecimentos	vii
Índice	ix
Lista de figuras	xi
Lista de tabelas	xiii
Abreviaturas e Símbolos.....	xv
Capítulo 1	1
Introdução	1
1.1 - Apresentação do problema	1
1.2 - Organização da dissertação	1
Capítulo 2	3
Laboratórios On-line	3
2.1 - Organização	3
2.2 – Diversidade.....	4
2.3 - Vantagens.....	6
Capítulo 3	7
Desenvolvimento de bancadas on-line com <i>NI ELVIS + LabView</i>	7
3.1 – Funcionalidades e <i>workflow</i> do <i>LabView</i>	8
3.2 – <i>NI ELVIS</i> : Apresentação e interação com o <i>LabView</i>	10
Capítulo 4	15
Especificação funcional de uma bancada on-line para a família 80C51	15
4.1 – Funcionalidade pretendida	15
4.2 – Interface com o utilizador.....	16
4.3 – Requisitos da ferramenta de reserva do acesso à bancada	18

Capítulo 5	19
Realização técnica	19
5.1 – Instalação do servidor Moodle e autenticação dos alunos.....	19
5.2 – Mecanismos de reserva de acesso	21
5.2.1 – Selecção	22
5.2.2 – Integração no Moodle	23
5.3 – Instalação do <i>software</i> e criação dos VI's.....	25
5.4 – <i>Hardware</i> da bancada on-line	29
5.4.1 – A carta CORE51 e a ligação à estação NI ELVIS	29
5.4.2 – Adaptação a outros Microcontroladores/Microprocessadores	30
Capítulo 6	33
Apresentação da bancada	33
6.1 – Autenticação e reserva da bancada	33
6.2 – Interface com a bancada	35
6.3 – Exemplo de aplicação	37
Capítulo 7	43
Conclusão	43
7.1 – Análise crítica.....	43
7.2 – Direcções de futuro	44
Referências	45
Anexo A: Alterações no bloco MRBS	47
Anexo B: Enunciado do trabalho EEC0029 relativo ao dado electrónico	59

Lista de figuras

Figura 2.1 - Estrutura de um laboratório on-line [origem: 2].....	4
Figura 2.2 - Estrutura dos laboratórios virtuais	5
Figura 2.3 - Estrutura dos laboratórios mistos	5
Figura 3.1 - Conjunto NI ELVIS + LabView + DAQ + Placa de protótipo [origem: 4]	8
Figura 3.2 - Painel frontal e diagrama de blocos do VI de comunicação	9
Figura 3.3 - Selecção do VI a publicar e activação do <i>WebServer</i>	11
Figura 3.4 - URL de acesso ao painel de controlo via Web.....	12
Figura 3.5 - Interface do VI acedido através de um navegador de internet	13
Figura 4.1 - Diagrama da blocos da bancada on-line para a família 80C51	16
Figura 5.1 - Vista semanal da ferramenta de reserva (MRBS)	23
Figura 5.2 - VI de interacção com a <i>CORE51</i>	26
Figura 5.3 - VI de instrumentação	27
Figura 5.4 - VI de comunicação	28
Figura 5.5 - Carta <i>CORE51</i> [origem: 8]	30
Figura 5.6 - Placa Keil MCB900 [origem: 9].....	31
Figura 6.1 - Confirmação da reserva.....	34
Figura 6.2 - Visualização da reserva	34
Figura 6.3 - Selecção da experiência no caso de multi-laboratório	35
Figura 6.4 - Página inicial do Moodle da FEUP na sua versão de testes	35
Figura 6.5 - Janela Web do VI de comunicação	36
Figura 6.6 - Janela Web do VI de interacção com a experiência	37
Figura 6.7 - Página inicial do Moodle de testes	38

Figura 6.8 - Página da reserva	39
Figura 6.9 - Janelas de navegação com os dois VI's carregados	39
Figura 6.10 - Transferência do código hexadecimal	40
Figura 6.11 - Lançamento do dado em que saiu o número dois	41
Figura 6.12 - Lançamento do dado em que saiu o número três	41

Lista de tabelas

Tabela 5.1 - Modos de operação da Keil MCB900 [origem: 8]	32
Tabela 9.1 - Configuração dos períodos de reserva	47
Tabela 9.2 - Configuração inicial do bloco MRBS	48
Tabela 9.3 - Remoção dos campos desnecessários.....	49
Tabela 9.4 - Lançamento da hiperligação de acesso aos painéis de controlo.....	52
Tabela 9.5 - Compatibilização dos dados temporais.....	53
Tabela 9.6 - Endereços de acesso aos VI's.....	56
Tabela 9.7 - Ficheiro comunicacao.PHP	56
Tabela 9.8 - Ficheiro controlo.PHP.....	57

Abreviaturas e Símbolos

Lista de abreviaturas (ordenadas por ordem alfabética)

ABCM	Associação Brasileira de Ciências Mecânicas
ASCII	<i>American Standard Code for Information Interchange</i>
CC	Corrente Contínua
CA	Corrente Alternada
DAQ	<i>Data Acquisition</i>
DEEC	Departamento de Engenharia Electrotécnica e de Computadores
E/S	Entradas/Saídas
FEUP	Faculdade de Engenharia da Universidade do Porto
LDAP	<i>Lightweight Directory Access Protocol</i>
LED	<i>Light Emitting Diode</i>
MIEEC	Mestrado Integrado em Engenharia Electrotécnica e de Computadores
MRBS	<i>Meeting Room Booking System</i>
NI	<i>National Instruments</i>
PC	<i>Personal Computer</i>
PLCC	<i>Plastic Leaded Chip Carrier</i>
RAM	<i>Random Access Memory</i>
ROM	<i>Read Only Memory</i>
SIFEUP	Sistema de Informação da Faculdade de Engenharia da Universidade do Porto
USB	<i>Universal Serial Bus</i>
VI	<i>Virtual Instrument</i>

Capítulo 1

Introdução

Neste capítulo é apresentado o trabalho a desenvolver e é descrita a forma como está organizada a dissertação. A apresentação do problema é feita sem grandes pormenores, visto que nos capítulos seguintes todos os assuntos referentes à dissertação são abordados com maior profundidade. A apresentação tem como objectivo dar a conhecer ao leitor, de forma superficial, os assuntos tratados neste documento.

Esta dissertação foi enquadrada pelo projecto POCI 2010 *Labs-On-The-Web*, que é financiado no âmbito do Programa Operacional Ciência e Inovação 2010 e pelo qual a FEUP é entidade responsável.

1.1 - Apresentação do problema

Pretende-se com esta dissertação, desenvolver uma bancada que permita aos alunos de Microprocessadores realizar experiências no âmbito da cadeira, a qualquer hora e a partir de qualquer local. Foi tomado como ponto de partida um protótipo já existente, que apenas permitia a realização de um leque muito reduzido de experiências. O acesso remoto a laboratórios pode ser dividido em duas etapas, uma de reserva do período de acesso e outra que diz respeito à interacção com a experiência.

1.2 - Organização da dissertação

No segundo capítulo é explicada a metodologia de funcionamento dos laboratórios on-line, as suas vantagens face a outro tipo de métodos de experimentação laboratorial e também a sua diversidade de utilização, tanto no meio pedagógico, como no meio industrial.

Uma vez que o *software* e o *hardware* de aquisição e controlo de dados foram suportados pelo conjunto LabView e ELVIS da National Instruments, no terceiro capítulo é abordada a metodologia de desenvolvimento de bancadas on-line com este conjunto. Esta tecnologia tem vindo a ser cada vez mais popular no seio das áreas técnicas do ensino secundário e superior.

Pretende-se que a solução final permita a realização de experiências no âmbito da cadeira de Microprocessadores. Uma vez que o microcontrolador usado e estudado nesta cadeira é da família 80C51, no capítulo quatro são descritas as funcionalidades proporcionadas por este

microcontrolador. Para além deste ponto, são abordados ainda no mesmo capítulo os requisitos da ferramenta de reserva do acesso à bancada, bem como a interface pretendida com o utilizador.

O quinto capítulo é de carácter técnico e aborda assuntos como a integração da ferramenta de reserva no servidor Moodle e as suas alterações, de modo a que cumpra os requisitos pretendidos. Para além do mecanismo de reserva, nesse capítulo é também esclarecido o processo de criação dos VI's que permitem o controlo das experiências, bem como a instalação do seu *software* de suporte.

O sexto capítulo apresenta uma componente pedagógica, nomeadamente a descrição do mecanismo de reserva da bancada e de controlo da experiência, tudo sob o ponto de vista do utilizador.

A conclusão é efectuada no capítulo sete, onde são analisados de forma crítica os resultados obtidos e onde são enumeradas possíveis direcções de futuro a tomar por este trabalho.

Seguem-se as referências bibliográficas e os anexos, que fecham este documento.

Capítulo 2

Laboratórios On-line

Neste capítulo abordamos o conceito de laboratórios on-line quanto à sua organização, às vantagens e às diversas aplicações actualmente suportadas por esta tecnologia.

O aumento de velocidade na transmissão de dados via Web e a utilização cada vez mais alargada da Internet, têm motivado o aparecimento, um pouco por todo o mundo, de um número cada vez maior de experiências que podem ser acedidas remotamente – correntemente designados por laboratórios on-line [1].

Os laboratórios remotos são uma nova tendência ao nível da aprendizagem, que se caracteriza pela forte componente de educação à distância e que permite derrubar barreiras geográficas. Estes laboratórios são uma das aplicações do *e-learning* e são cada vez mais as escolas de formação e universidades que adoptam esta tecnologia para melhorar o sucesso escolar dos seus alunos.

2.1 - Organização

O conceito base de laboratório on-line pode ser dividido em três partes: o cliente, o servidor e a internet/LAN. No próximo capítulo abordaremos ao pormenor os periféricos e o sistema utilizado neste trabalho em particular.

A figura 2.1 ilustra a organização de um laboratório on-line e a interacção entre os seus vários elementos. A experiência encontra-se ligada ao servidor de laboratório (computador) que implementa as conexões necessárias com a plataforma que vai suportar a experiência. O acesso ao servidor de laboratório pode ser feito directamente pelo aluno através da internet ou rede LAN da instituição, ou através de um servidor de *e-learning*. No nosso caso o acesso é efectuado por este segundo método, mais concretamente através do Moodle. De salientar que o acesso pode ser efectuado a partir de qualquer local, ou seja, o aluno pode estar ligado à rede da instituição ou em qualquer outro local com acesso a uma ligação de internet. Apesar de a plataforma onde se encontra a experiência se encontrar fisicamente perto do servidor de laboratório, o mesmo já não acontece com o servidor de *e-learning*, que pode estar em qualquer ponto da instituição. Isto deve-se ao facto de o servidor de *e-learning* ter como principal objectivo proporcionar o enquadramento técnico e o lançamento da hiperligação de

acesso à experiência, o que torna evidente a não obrigatoriedade destes dois elementos se encontrarem próximos.

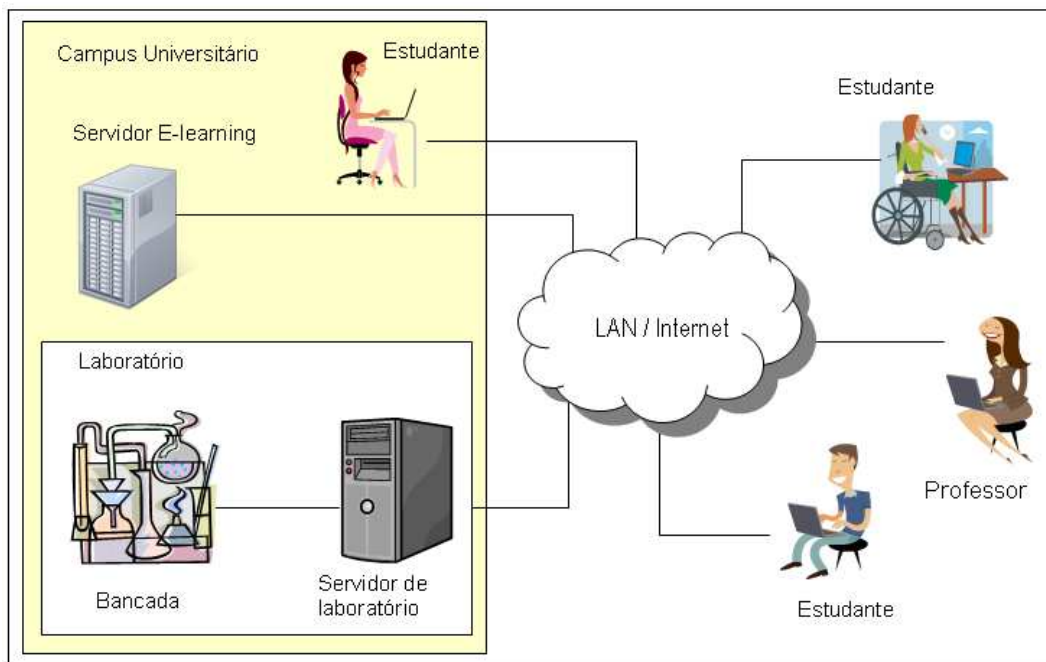


Figura 2.1 - Estrutura de um laboratório on-line [origem: 2]

Outra componente importante no acesso a laboratórios on-line é a ferramenta de reserva que permite aos utilizadores marcarem a hora do seu acesso ao laboratório on-line e também terem conhecimento do actual preenchimento deste calendário de reservas. Esta ferramenta é instalada no servidor de *e-learning*, neste caso o servidor Moodle da FEUP, devido à facilidade de acesso e autenticação via *LDAP* actualmente suportada por este servidor. No capítulo quatro abordaremos a ferramenta escolhida para este efeito. Quanto às alterações no código fonte, de modo a tornar esta ferramenta o mais compatível possível com os requisitos pretendidos, serão abordadas no capítulo cinco.

2.2 – Diversidade

O interesse da utilização destas tecnologias é muito alargado, não se restringindo apenas ao ensino universitário. Exemplo disso é o caso das aplicações em medicina (apoio ao diagnóstico e às intervenções cirúrgicas) e à área da manutenção industrial, tornando-se também facilmente extensível a actividades de investigação e desenvolvimento [1].

Os laboratórios on-line podem localizar-se tanto nas instalações da faculdade como fora delas. Vários departamentos da FEUP possuem experiências acessíveis remotamente e situadas nas suas instalações. Como exemplo do acesso on-line a uma experiência que se situa fora das instalações da FEUP, pode citar-se o teste de memórias SDRAM, cujo equipamento está situado na empresa Qimonda. Devido à impossibilidade financeira de suportar o elevado custo deste equipamento, a FEUP e a Qimonda tem um acordo que permite

aos alunos da cadeira de Testes de Sistemas Electrónicos acederem remotamente a este equipamento para realizarem os testes pretendidos.

Quanto às áreas disciplinares, o curso de engenharia electrotécnica e computadores e o curso de engenharia mecânica, apresentam maior adesão a este tipo de tecnologias. Isto deve-se à variedade de experiências possíveis de desenvolver no âmbito das cadeiras, pois apresentam maior facilidade de as implementar em laboratórios e torná-las acessíveis remotamente. Esta facilidade deve-se ao actual suporte a nível de *software* e *hardware*, que permite implementar diversas experiências em ambiente laboratorial.

Os laboratórios em que não é necessária a presença do aluno podem ser de natureza remota, virtual e mista. Os laboratórios por nós abordados são do primeiro tipo (figura 2.1), onde a bancada de trabalho existe fisicamente, sendo o acesso e o controlo efectuados remotamente através de uma rede LAN ou internet.

Os laboratórios virtuais (figura 2.2), baseados em simuladores, não requerem necessariamente acesso à internet ou a uma rede LAN, uma vez que o *software* de simulação pode ser instalado em qualquer computador de laboratório ou de uso pessoal. Isto torna este tipo de solução muito utilizada para preparação de aulas práticas, uma vez que é acessível a todos os alunos e que lhes permite simular e efectuar os cálculos necessários para uma posterior aplicação na experiência real.

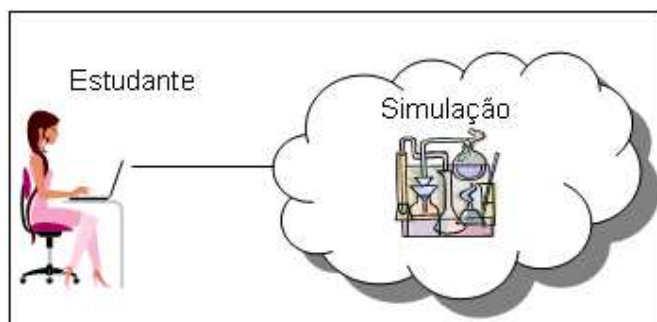


Figura 2.2 - Estrutura dos laboratórios virtuais

Já os laboratórios mistos (figura 2.3) são uma fusão destes dois últimos, pois apresentam uma componente física e uma componente de simulação, sendo o acesso feito remotamente através da internet ou rede LAN. Este tipo de laboratórios justifica-se pela facilidade de simulação de certas experiências (ou parte delas) face à sua montagem física e por motivos de ordem financeira.

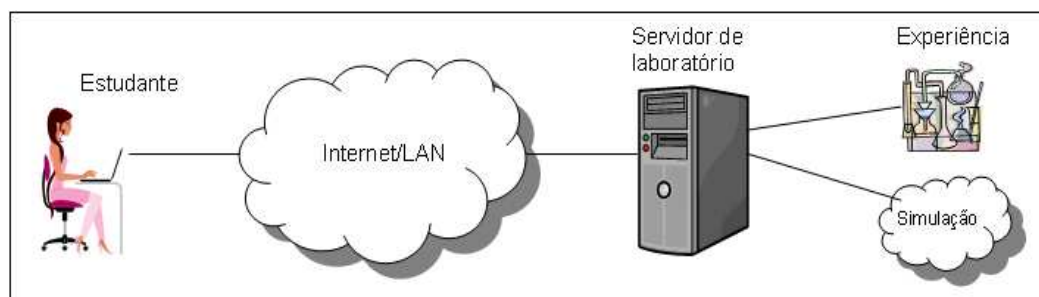


Figura 2.3 - Estrutura dos laboratórios mistos

2.3 - Vantagens

Quando se fala de laboratórios on-line e na sua utilização no ensino, a primeira questão que é levantada é sobre as vantagens proporcionadas por esta tecnologia. É importante mencionar que não é intenção deste método de ensino substituir a habitual relação pedagógica professor - aluno e a importante actividade presencial de laboratório, pois certas competências só são adquiridas em contacto directo com o equipamento. A comunicação cara-a-cara nunca deverá perder a sua importância e lugar na educação, mas não há dúvida que estas iniciativas e inovações são de louvar.

Em muitas áreas, o recurso a estas tecnologias apresenta para o ensino tradicional e para a actualização ao longo da vida importantes aspectos de complementaridade na formação. Mais ainda, acrescenta um conjunto de outras vantagens, como a utilização dos laboratórios em horário alargado e sem a presença de um técnico. Proporciona também a partilha de equipamento com outros laboratórios ou instituições, e o acesso a equipamento dispendioso ou cuja manipulação possa representar algum risco. O acesso simplificado para utilizadores com limitações físicas ou logísticas [2] proporciona uma excelente alternativa de ensino e atende ao ritmo de aprendizagem de cada aluno, combatendo assim o insucesso escolar.

Os alunos que por algum motivo não possam comparecer às aulas práticas, ou que não tenham tido tempo de finalizar o trabalho durante o tempo da aula, podem agora aceder à experiência à hora que quiserem e a partir do local que quiserem.

No caso do ensino universitário, os sistemas que permitem o estudo de controladores e compensadores, em geral, são de alto custo e cada módulo admite apenas um utilizador ou um pequeno grupo de utilizadores a executar uma determinada experiência num dado instante. Dada a escassez de recursos, qualquer esforço para maximizar a utilização dos equipamentos disponíveis possibilita uma melhoria no aproveitamento do curso pelos alunos.

Face aos laboratórios virtuais, o acesso on-line apresenta um maior teor prático e realístico, bem como o acesso a experiências impossíveis de implementar em ambientes de simulação.

Capítulo 3

Desenvolvimento de bancadas on-line com *NI ELVIS + LabView*

Os laboratórios on-line permitem o controlo e a interacção com dispositivos físicos por meio de *software* e podem ser considerados como uma sofisticada plataforma interactiva de demonstrações. Se o acesso a estas bancadas for muito detalhado, pode ser um bom complemento para um laboratório real, especialmente se acompanhado de animação. Estes laboratórios acessíveis através da Internet estão a transformar-se em uma maneira popular de reduzir custos de equipamento e disponibilizar conceitos laboratoriais em cursos de educação à distância. Estes tipos de laboratórios usam geralmente *softwares* comerciais como: *LabView*, *MATLAB*, *Arena*, *AutoMod*, entre outros.

O mundo académico, mais especificamente nas áreas da engenharia e das ciências, estão comprometidas com a National Instruments, que procura desenvolver *software* e *hardware* flexíveis e robustos compatíveis com as tecnologias baseadas em PC, permitindo a fusão de conhecimentos teóricos com aplicações reais. O conceito de instrumentação virtual, ou seja, a utilização do computador combinado com *software* de forma a implementar aplicações reais já é largamente utilizado tanto por engenheiros e cientistas, bem como professores e estudantes de todo o mundo. O impacto deste tipo de instrumentação, mais particularmente o *software* NI LabView é visível no mundo académico, onde professores, investigadores e alunos utilizam esta tecnologia para auxiliar o ensino. Os professores podem recorrer à integração de *hardware* e *software* da NI para ensinar uma ampla variedade de conceitos em áreas de aplicação como medições, circuitos, controlo, processamento de sinais e imagem, comunicações e sistemas embarcados [3].

No caso deste trabalho é utilizado o conjunto LabView + NI ELVIS, ambos da National Instruments. A plataforma NI ELVIS é uma bancada que suporta variados instrumentos de medida e de instrumentação, frequentemente utilizados em trabalhos práticos da cadeira de Microprocessadores. Em conjunto com o LabView, esta bancada permite ao utilizador controlar estes diversos componentes. A NI ELVIS permite a integração de uma placa que para além das diversas interfaces com os aparelhos de instrumentação já mencionados, também possibilita ao utilizador desenvolver os seus próprios circuitos numa área de protótipo. Na figura 3.1 é apresentada uma imagem do pacote da NI constituído pela NI ELVIS, LabView, DAQ e placa de protótipo.

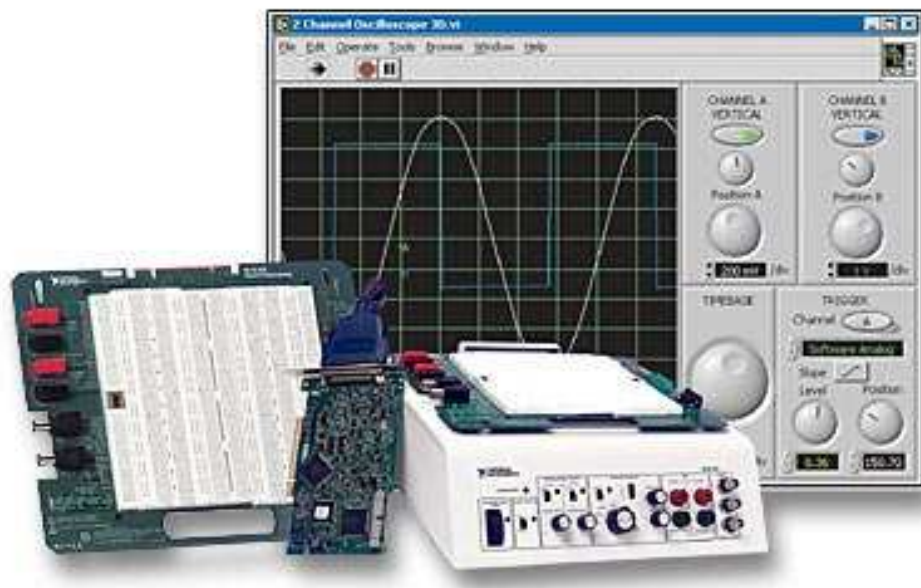


Figura 3.1 - Conjunto NI ELVIS + LabView + DAQ + Placa de protótipo [origem: 4]

3.1 – Funcionalidades e *workflow* do LabView

Hoje em dia, são inúmeras as faculdades, instituições e empresas a apostar no acesso e controlo remoto de experiências ou processos. A nível do ensino, o *software* LabView, da National Instruments, é uma das soluções com mais êxito e utilização. O LabView é uma ferramenta que possibilita construir painéis de controlo, chamados VI's que permitem ao utilizador visualizar e/ou controlar uma determinada experiência ou processo, tais como:

- Design:
 - Processamento de sinal e imagem;
 - Sistemas de controlo embebidos (PC, DSP, FPGA, Microcontroladores);
 - Simulação e protótipo;
- Controlo:
 - Controlo automático e sistemas dinâmicos;
 - Mecânica e robótica;
- Medidas:
 - Circuitos e electrónica;
 - Medidas e instrumentação;

Os VI's são painéis frontais que permitem controlar por meio de software um conjunto de instrumentos ou hardware que no nosso caso encontram-se na plataforma NI ELVIS.

A instrumentação virtual é utilizada em muitos tipos de aplicações, começando pela concepção, protótipo e desenvolvimento. A plataforma LabView proporciona ferramentas e modelos específicos para o seu desenvolvimento, com um intuitivo e poderoso paradigma gráfico. Requer um computador equipado com o *software* LabView e o respectivo, *hardware*,

tal como placas de aquisição, que em conjunto implementam as funções de instrumentos tradicionais.

Os VI's representam uma fundamental mudança dos tradicionais sistemas de instrumentação centrados no *hardware*, para sistemas centrados no *software*. Estes sistemas baseados no *software* exploram as capacidades do computador, produtividade, visualização, capacidades de conexão dos computadores e das estações de trabalho.

Embora o computador e a tecnologia de circuitos integrados tenham sofrido um significativo avanço nas duas últimas décadas, o *software* oferece realmente a flexibilidade de realizar, nesta poderosa fusão com o *hardware*, a criação de instrumentos virtuais. Isto proporciona melhores percursos para a inovação e significativo redução de custos. Com os instrumentos virtuais, engenheiros e cientistas construíram sistemas de medida e automação que se adequam às suas necessidades, ao invés de ficarem limitados à função dos instrumentos tradicionais.

O *software LabView* apresenta uma vasta biblioteca de funções de aquisição, controlo e visualização, que tornam o processo de criação de um VI menos complexo. O método de construção de um VI é dividido em duas partes, que compreendem o painel frontal e o bloco de diagramas, tal como mostra a figura 3.2.

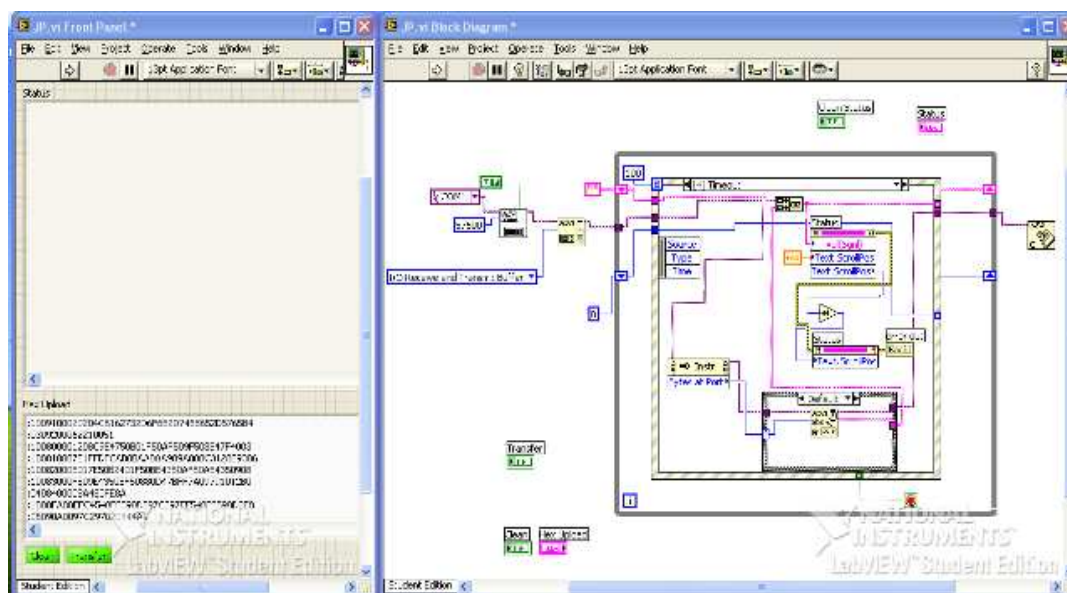


Figura 3.2 - Painel frontal e diagrama de blocos do VI de comunicação

O painel de controlo é a componente gráfica do VI, que não é mais do que a interface gráfica que permite ao utilizador visualizar e controlar a experiência.

O bloco de diagramas é a parte que integra todo o código de programação em *LabView*, ou seja, é a componente de programação que se encontra mediante o painel de controlo e que implementa todas as funções por ele desencadeadas. De salientar que o tipo de programação em *LabView* é gráfica e não textual, como sucede com a generalidade das linguagens de programação a que estamos habituados a utilizar. Como é óbvio, estas duas partes são construídas simultaneamente, pois as entradas e saídas dos diagramas de blocos são controlos e indicadores do painel frontal.

O passo seguinte, após ter o instrumento virtual concluído, é efectuar a sua interacção com a plataforma NI ELVIS e proceder à publicação do painel, de modo a que a experiência seja acessível via Web. No próximo subcapítulo exemplificamos este processo com uma pequena experiência que visa apenas ajudar a compreender o mecanismo de publicação de VI's com o conjunto LabView + NI ELVIS.

3.2 – NI ELVIS: Apresentação e interacção com o LabView

A plataforma NI ELVIS é um hardware que permite o desenvolvimento de protótipos baseados na tecnologia LabView para utilização em laboratórios. Com a utilização da ELVIS, os professores podem por ao dispor dos estudantes a tecnologia necessária para aprenderem a teoria e ao mesmo tempo pô-la em prática nas mais diversas áreas de utilização desta plataforma. As áreas dos circuitos electrónicos, processamento de sinais, comunicação, sistemas de controlo, medições mecânicas e mecatrónicas [5].

A NI ELVIS é composta por variados equipamentos de instrumentação e medida, um dispositivo multifuncional para aquisição de dados e uma placa para a criação e desenvolvimento de protótipos. A sua integração com o LabView permite o acesso a um conjunto de tradicionais instrumentos de laboratório, bem como o seu controlo via *software*. Dentro deste variado leque de instrumentos destacam-se pela sua importância de utilização na disciplina de Microprocessadores, o osciloscópio, gerador de funções, multímetro digital, fonte de alimentação programável e analisador de sinal. O facto de ser baseada em LabView e permitir a aquisição completa de sinais, acrescido da possibilidade de criação de protótipos, torna este sistema ideal para integrar instrumentação virtual nos cursos académicos, desde os técnicos até aos pós-graduados e mestrados [5]. De salientar que o *software* de instalação do NI ELVIS já inclui alguns VI's de extrema utilidade e previamente definidos para controlar o conjunto de instrumentos já mencionados.

Esta plataforma facilita aos estudantes a construção de circuitos e interfaces personalizadas. Utilizando uma placa removível para criação de protótipos, os estudantes podem desenhar seus próprios circuitos electrónicos, instrumentos de condicionamento de sinal e pequenos dispositivos electromecânicos. A placa para criação de protótipos vem com conectores tipo banana, BNC e D-sub para conexão fácil e segura. A estação de trabalho NI ELVIS também é equipada com uma protecção para curto-circuito e sobretensões para o dispositivo de aquisição de dados [5].

A fim de melhor compreender o mecanismo de publicação de VI's com o conjunto *LabView* + *NI ELVIS*, fizemos uma simples experiência que tinha como principal objectivo o acesso e controlo remotos de um VI, ou seja, através de um navegador com ligação à internet. O *software* da *NI ELVIS* engloba uma série de VI's de grande utilidade, sendo desnecessária a perda de tempo nas suas construções. Neste subcapítulo, o principal objectivo passa pela familiarização com o método de publicação e acesso online a um VI, deixando a construção dos VI's finais para os capítulos posteriores. Para tal foi utilizado um pequeno circuito amplificador, em que apenas dispomos de um osciloscópio e de um gerador de sinais.

O processo de publicação é de extrema facilidade, pois o *LabView* apresenta uma ferramenta chamada *Web Publishing Tool* (figuras 3.3 e 3.4), em que apenas temos que seleccionar o modo de visualização e o título que irá ser apresentado na página de navegação quando o utilizador aceder ao VI.

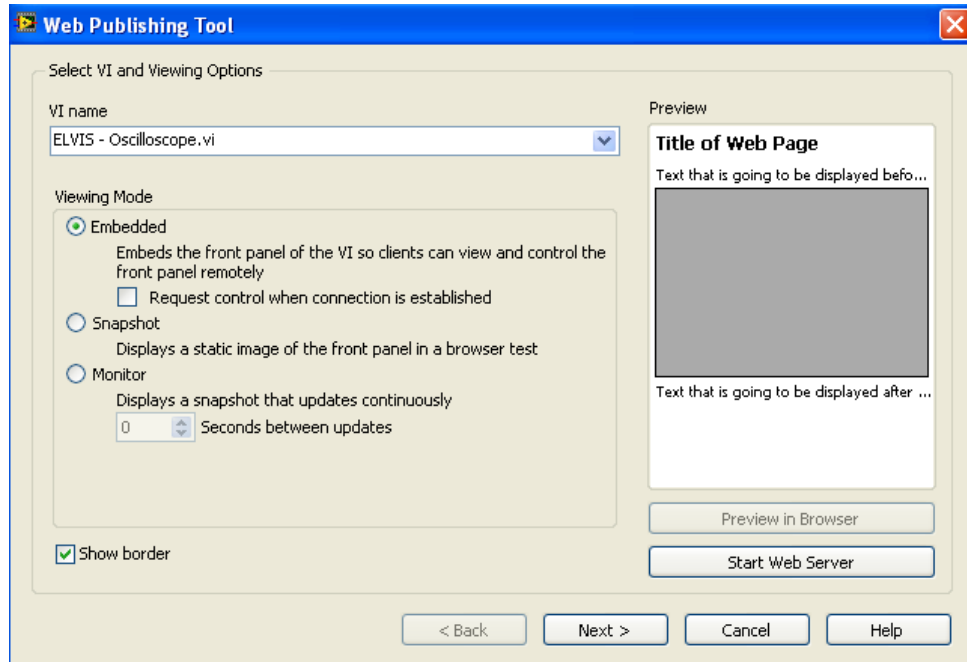


Figura 3.3 - Selecção do VI a publicar e activação do *WebServer*

Foi utilizado um circuito integrado TL082, um conhecido dispositivo que contém dois amplificadores operacionais. Para além do amplificador, foram também usadas duas resistências que definem o ganho do conjunto. Uma vez montado o circuito na placa da plataforma NI ELVIS, tivemos que proceder à publicação dos VI's necessários a esta experiência, que integram um osciloscópio e um gerador de sinais. Este processo, tal como referido anteriormente, não apresenta elevado grau de complexidade, bastando apenas fazer a publicação dos VI's, tal como se ilustra na figura 3.4 que devem permanecer em modo de execução no servidor de laboratório. Os laboratórios são desenvolvidos utilizando painéis de controlo que residem no computador local, coordenando a experiência. Na figura 3.3 encontram-se ilustrados os modos de visualização dos VI's, como por exemplo, permitir o controlo por parte do utilizador que acede via Web à experiência. Outras opções disponíveis são a alteração do título da página, do texto que vai constar acima e abaixo do VI. O endereço de acesso via internet à experiência encontra-se representado no campo URL da figura 3.4.

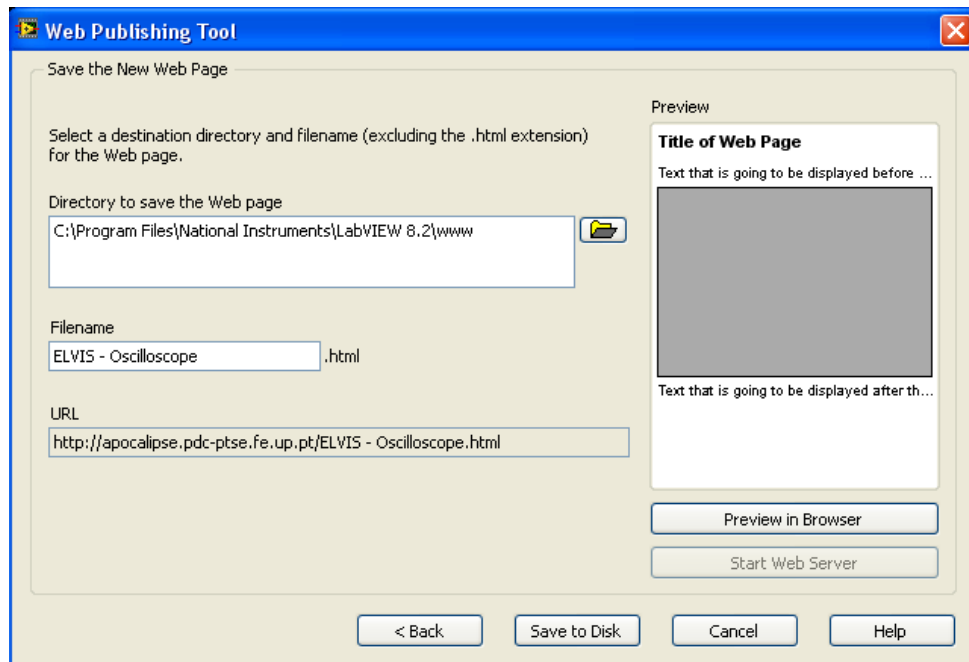
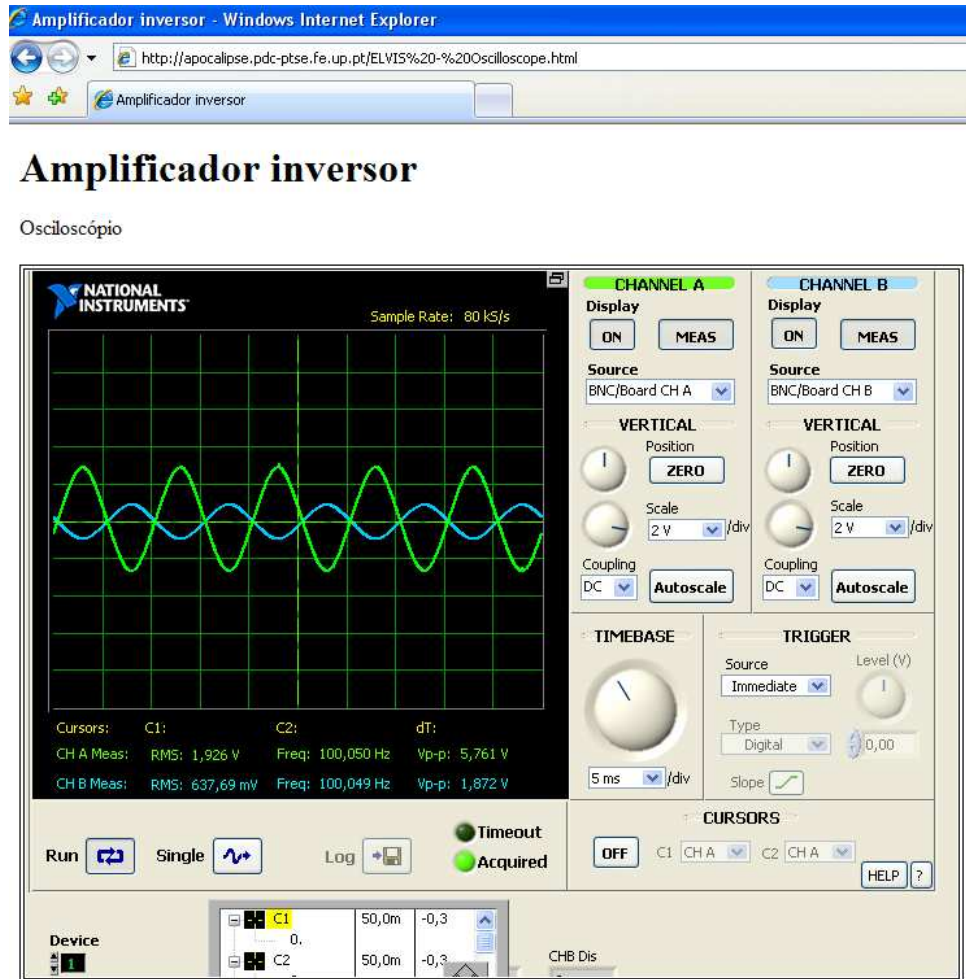


Figura 3.4 - URL de acesso ao painel de controlo via Web

Conceptualmente, usar um browser como interface para o laboratório on-line tem muitas vantagens em conexões Intranet e/ou Internet. É uma plataforma independente e fácil de usar e o *software* adicional necessário no lado do aluno (utilizador remoto) é mínimo. Entretanto, a conexão Web também fornece alguns desafios: necessita-se de meios para que o estudante incorpore os parâmetros do controlo, que usem preferivelmente uma interface gráfica amigável, e de meios para simular a resposta do sistema [6].

Uma vez acedido ao endereço fornecido pelo *Web Publishing Tool* através do navegador, é carregada uma página com o aspecto da figura 3.5. De salientar, que neste exemplo só é ilustrada a publicação e o acesso ao osciloscópio, sendo idêntico o processo para o caso do gerador de sinais. Na figura 3.5 podemos verificar a existência de dois sinais, o de entrada (a azul) e o de saída (a verde), que apenas diferem em amplitude e fase, fruto do ganho aplicado pelo amplificador inversor.



Sinal na entrada e na saída do ampop

Figura 3.5 - Interface do VI acedido através de um navegador de internet

Capítulo 4

Especificação funcional de uma bancada on-line para a família 80C51

Neste capítulo são abordados os requisitos e funcionalidades de uma bancada on-line para a família 80C51. Para além disso é explicado o tipo de interface desejado na interacção com a bancada, no que respeita aos VI's. Por último são descritas as funcionalidades necessárias para a ferramenta de reserva da bancada, uma vez que as aplicações existentes para este fim podem não ir ao encontro dos requisitos da nossa bancada.

Porquê a utilização da família 80C51 e não de uma outra neste trabalho? A resposta deve-se ao facto de este microcontrolador ser o estudado e utilizado na cadeira de Microprocessadores para o desenvolvimento dos vários trabalhos práticos ao longo do semestre. Além disso, é um microcontrolador que, apesar da sua fácil compreensão e manuseamento, é avançado o suficiente ao ponto de permitir implementar com facilidade os trabalhos a realizar nesta disciplina.

4.1 – Funcionalidade pretendida

De modo a que uma bancada on-line para a família 80C51 seja o mais funcional possível, é necessário que o microcontrolador cumpra os requisitos para que a bancada se torne adequada ao ensino da disciplina de Microprocessadores.

Numa bancada para a família 80C51 é necessário ter um conjunto de entradas e saídas acessíveis ao utilizador, bem como interface que lhe permita programar o microcontrolador com o código desejado. Quanto ao tipo de interface com o servidor de laboratório, RS232C é o mais usual e de fácil utilização, sendo suportado pela família 80C51. É necessário um conjunto de entradas e saídas digitais acessíveis ao utilizador remoto, de modo a que possam ser controlados os mais diversos dispositivos. Não é de necessidade obrigatória, mas é no mínimo vantajoso, que para além das entradas digitais esta bancada possua também uma entrada analógica ou um conversor A/D. A família 80C51 cumpre todos estes requisitos, pois apresenta um conjunto de pinos que podem ser configurados tanto como entradas como saídas digitais. Para além disso, possui um conjunto de entradas que podem ser configuradas de modo a

termos disponível um conversor A/D. Por último, esta família permite ainda a ligação RS232C, o que admite uma fácil ligação ao servidor de laboratório.

A CORE51 permite uma melhor familiarização com os modos de funcionamento dos periféricos internos dos microcontroladores dessa família, bem como de muitos outros compatíveis com ela. Pode ser utilizada como um sistema autónomo para auto-aprendizagem ou como núcleo de circuitos mais complexos, dispondo para isso de todos os sinais necessários à sua expansão.

O desenvolvimento de programas para esta placa deve ser feito tendo em conta que serão executados a partir do endereço 0x0000. Podem ser desenvolvidos em *assembly* ou qualquer linguagem de alto nível para a qual exista compilador [5].

Para interface com o utilizador, dispõe de quatro LED's, activos a zero, ligados de P1.4 (LED 1) a P1.7 (LED 4), e de quatro teclas, activas a zero, ligadas de P1.3 (TECLA 1) a P1.0 (TECLA 4). Dispõe ainda de uma tecla (RESET) para inicialização manual do programa e outra (INTR) para geração de uma interrupção. Para proceder à ligação e teste da placa, esta deve ser ligada à saída de 12V da fonte de alimentação e a uma porta série do computador.

A integração da carta CORE51 nesta bancada torna-se extremamente útil e permite abranger todos os requisitos necessários. Assim que a CORE51 recebe o código Intel HEX para a sua programação, fica à espera do carácter que dá início à execução do código. A figura seguinte ilustra o diagrama de blocos do sistema baseado na tecnologia NI ELVIS + LabView para o desenvolvimento da bancada on-line para a família 80C51.

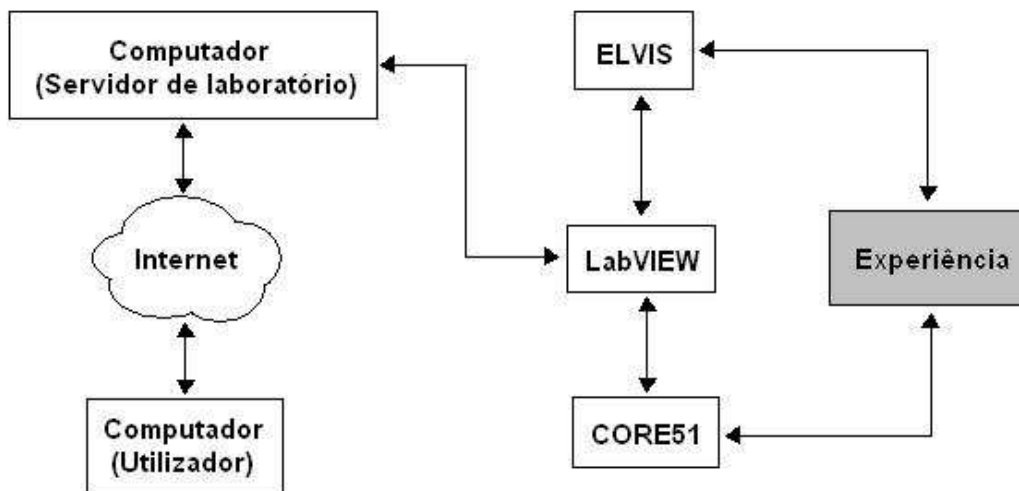


Figura 4.1 - Diagrama da blocos da bancada on-line para a família 80C51

4.2 – Interface com o utilizador

Deve ser possível ao aluno transferir o código Intel HEX das respectivas experiências para a memória do microcontrolador, via porta série, tal como efectuado nas aulas práticas com a utilização da carta CORE51. Assim sendo, é necessário integrar a CORE51 com o pacote LabView + NI ELVIS, de forma clara e objectiva, tornando possível a transferência do código remotamente. Este requisito implica a criação de um VI que possibilite a transferência do código para o microcontrolador, ou seja, um VI que implemente a comunicação entre a carta

CORE51 e o LabView. Este VI graficamente deve apresentar uma caixa de texto, bem como um botão que, quando pressionado, inicia a transferência do código existente na caixa de texto para a memória da CORE51. Para além disso, deve integrar uma outra caixa de texto que vai conter informação relativa à comunicação entre o *LabView* e a *CORE51*, ao correcto estado das memórias da CORE51, bem como ao seu modo de funcionamento.

De forma a controlar as quatro teclas da CORE51 e ainda a tecla de RESET e INTR é necessário um VI que permita comandar estas entradas através da utilização de controladores booleanos. Para além disso, deve ser ainda possível neste *VI* efectuar a escolha do modo de funcionamento da CORE51, ou seja, escolher entre os modos de *cold test*, *hex upload* e *manual*.

Assim que seleccionado o modo *cold test*, activam-se sequencialmente activas as entradas controladas pelas teclas da CORE51, que envia para a caixa de texto do VI de comunicação o estado das memórias. Já quando é escolhido o modo *hex upload*, é enviada novamente para a mesma caixa de texto a indicação do estado das memórias, bem como a indicação de que a CORE51 se encontra à espera da transferência do código hexadecimal. Por último, o modo *manual*, permite ao utilizador correr o programa transferido para a memória da CORE51 e também controlar as teclas que comandam as suas entradas. Posteriormente este VI vai ser referido como o VI de interacção com a carta, uma vez que é através dele que é feita a selecção do modo de funcionamento da CORE51 e que são controladas as suas teclas, assim que é iniciada a execução do código.

Torna-se evidente a necessidade de interacção entre o VI de controlo e o de comunicação, uma vez que a escolha do modo de funcionamento vai influenciar a informação a ser mostrada na caixa de texto do VI de comunicação.

O osciloscópio e o gerador de funções são dois instrumentos de extrema importância na área de electrónica e consequentemente em Microprocessadores, com os quais estamos muito familiarizados. Uma vez que o objectivo deste trabalho é a implementação de uma bancada que permita o ensino de Microprocessadores, a ausência destes dois instrumentos reduziria em grande escala a variedade de experiências possíveis de implementar. Assim sendo, é necessário o desenvolvimento de um ou dois VI's que implementem o controlo do osciloscópio e do gerador de funções integrados na plataforma NI ELVIS.

A existência de experiências que necessitam de ser seguidas on-line através da publicação de imagens em tempo real (vídeo ou fotografias) na interface com o utilizador, obriga à integração da imagem adquirida por uma câmara USB. Exemplos deste tipo de experiências são o caso do dado electrónico e do controlo da intensidade luminosa de uma lâmpada. Assim sendo é primordial a introdução da imagem adquirida pela câmara em algum dos VI's acima referidos, ou então a criação de um VI individual, que permita a publicação dessa imagem.

Tendo em conta o já numeroso número de VI's necessários de implementar, é levantado o problema do lançamento de demasiadas janelas Web, cada uma associada a um VI. Desta forma é de boa prática procurar uma solução que permita reduzir ao máximo o número de painéis a publicar, tornando a interacção com a experiência o mais eficaz possível.

No capítulo cinco voltaremos a falar sobre estes VI's, mas desta feita será do ponto de vista da sua criação e publicação, e será também abordada a problemática do lançamento de demasiados painéis de controlo.

4.3 – Requisitos da ferramenta de reserva do acesso à bancada

Como já foi descrito anteriormente, o acesso a uma bancada on-line requer um método de reserva por parte dos alunos, de forma a tornar esse acesso o mais organizado e proveitoso possível. É necessário que esta ferramenta cumpra certos requisitos funcionais, de forma a tornar clara e objectiva a sua utilização. Os requisitos vão ser divididos quanto às funções e quanto à forma que a ferramenta deve implementar.

No que respeita às funcionalidades, é inevitável que esta ferramenta possa ser integrada num servidor de *e-learning*, neste caso o Moodle da FEUP. Uma vez que uma bancada só pode ser acedida por um utilizador de cada vez, deve permitir-se unicamente uma reserva por cada período horário. Deve ser possível ao utilizador que a efectuou anulá-la ou editá-la sempre que desejado, assim como a edição e remoção de qualquer reserva por parte de um utilizador com permissão para tal, neste caso o professor. O professor deve ter ainda a possibilidade de ser notificado por e-mail quando ocorrer a adição, edição ou remoção de uma reserva. Após chegada a hora e o dia escolhidos, deve ser lançado na respectiva página uma hiperligação de acesso aos painéis de controlo da experiência. Notar que esta hiperligação só deve ser visível quando a página é acedida pelo utilizador que efectuou a reserva e durante o período reservado.

Quanto à forma, é pretendido que o interface primário seja um calendário com vista semanal, em que cada dia é dividido em períodos de uma hora. Os utilizadores que acederem a esta ferramenta devem poder visualizar o preenchimento actual de reservas de acesso ao laboratório, bem como os utilizadores que as efectuaram.

Capítulo 5

Realização técnica

Os capítulos anteriores são de um teor maioritariamente teórico, em que é abordada a temática do acesso e controlo de laboratórios on-line, introduzindo conceitos e metodologias actualmente utilizadas neste tipo de ensino. Com este capítulo pretendemos apresentar as escolhas e implementações efectuadas no decorrer deste trabalho, ou seja, apresenta-se uma abordagem do ponto de vista técnico, que serve também como guia de utilização para esta bancada.

5.1 – Instalação do servidor Moodle e autenticação dos alunos

Uma vez que se pretende integrar a ferramenta de acesso no servidor de *e-learning* da FEUP, foi instalado um servidor Moodle num computador do laboratório, onde decorreu este trabalho. O objectivo desta instalação foi permitir que fossem feitas as devidas alterações na ferramenta de reserva, de modo a esta cumprir os requisitos e poder ser testada constantemente, sem com isso pôr em causa o correcto funcionamento do servidor Moodle da FEUP. Desta forma, foi possível efectuar alterações regulares no código da ferramenta de reserva escolhida e observar os resultados de forma imediata. Todo este processo torna-se transparente, na medida em que o servidor de *e-learning* instalado no laboratório é o mesmo que o da faculdade, bastando unicamente transferir o bloco para o Moodle da FEUP, assim que foi concluído.

Para ser criado um servidor Moodle são necessários três requisitos fundamentais: um servidor Web que suporte PHP (por exemplo Apache), um servidor de base de dados (por exemplo o MySQL ou PostgreSQL) e a linguagem PHP. A forma mais fácil de instalar tudo o que é necessário é usar o pacote “EasyPHP”, que inclui todo este *software* numa única aplicação. Os passos a seguir são os seguintes [7]:

1. Primeiro, se já foi instalado o MySQL anteriormente (em forma isolada ou como parte de outro pacote), é necessário desinstalá-lo e apagar todos os ficheiros do MySQL.
2. Se alguma vez já foi instalado o PHP, é necessário apagar do directório do Windows, os ficheiros com nome “php4ts.dll” e “php.ini”.

3. É necessário obter uma cópia do EasyPHP, que uma vez que é software gratuito, pode ser encontrado com uma simples pesquisa na internet.
4. Executar a aplicação "easyphpsetup.exe". É sugerido aceitar todos os valores predefinidos e avançar pelos passos do processo de instalação.
5. No fim do processo de instalação, deixar seleccionada a caixa que diz "Iniciar EasyPHP" e carregar no botão "Terminar".
6. Se tudo correr bem o servidor Apache, PHP e MySQL já estarão instalados e a funcionar. Deve ser visualizado um "E" preto na barra de ferramentas. É possível carregar nele com o botão direito do rato para aceder a um menu que permite controlar os programas em execução.
7. O próximo passo é preparar uma base de dados a ser usada pelo Moodle. Para tal deve-se carregar no E preto, na barra de ferramentas, com o botão direito do rato, seleccionar Administração e carregar sobre DB Management (a lado de PHPMyAdmin);
8. Quando for pedido um nome de utilizador, usar "root" com palavra-chave em branco. Deve ser visualizada uma interface do phpMyAdmin que permite criar novas bases de dados e contas de utilizadores.
9. De seguida iremos criar uma nova base de dados, onde escrevemos "Moodle" no campo do nome e carregamos no botão "Create"
10. Já temos o que precisamos para instalar o Moodle. É necessário obter uma cópia da sua versão mais recente a partir do endereço "http://moodle.org/download" e descompactar o ficheiro zip descarregado.
11. De seguida copiamos os ficheiros do Moodle para "C:\Program Files\EasyPHP\www". Podemos copiar o directório completo (por exemplo "C:\Program Files\EasyPHP\www\moodle") ou todo o conteúdo do directório Moodle. Se optarmos pela segunda escolha, devemos poder aceder à pagina do Moodle através do endereço "http://localhost/", em vez de "http://localhost/moodle/".
12. De seguida é necessário criar um novo directório vazio, em local à escolha, para onde o Moodle copiará os ficheiros recebidos; por exemplo "C:\moodledata".
13. Entramos no directório do Moodle e fazemos uma cópia de config-dist.PHP e mudamos-lhe o nome para config.PHP.
14. Editamos o ficheiro config.PHP usando algum editor de texto (ter sempre o cuidado de não introduzir espaços em branco adicionais no fim de cada linha).
15. Escrevemos a informação sobre a nova base de dados:
\$CFG->dbtype = 'mysql';
\$CFG->dbhost = 'localhost';
\$CFG->dbname = 'moodle';
\$CFG->dbuser = 'root';
\$CFG->dbpass = '';
\$CFG->dbpersist = true;
\$CFG->prefix = 'mdl_'
16. Escrevemos também o caminho completo dos seus ficheiros:
\$CFG->wwwroot = 'http://localhost/moodle'; // Podemos usar endereço externo no caso de termos algum.

```
$CFG->dirroot = 'C:\Program Files\EasyPHP\www\moodle';
```

```
$CFG->dataroot = 'C:\moodledata'
```

17. Gravamos o novo ficheiro config.php.
18. A instalação está quase concluída e o resto do processo é feito através da página Web. Acedemos à página “http://localhost/moodle/admin/” com o navegador Web, para continuar com o processo de instalação.
19. Carregamos com o botão direito do rato sobre o “E” preto na barra de ferramentas para aceder o menu e seleccionamos “Restart”.

Alguns dos módulos do Moodle precisam de verificações frequentes para realizar algumas tarefas. Por exemplo, o Moodle precisa verificar os fóruns de discussão para saber se é preciso enviar por correio cópias de novas contribuições aos assinantes do fórum. O script que executa essas tarefas de rotina encontra-se no directório “*admin*”, com o nome “*cron.php*”. No entanto, ele não pode arrancar por si próprio, sendo preciso instalar um mecanismo para que o *script* seja executado a intervalos regulares (por exemplo, cada 5 ou 10 minutos). Esse mecanismo constitui as “pulsações cardíacas” necessárias para que o *script* possa executar as tarefas definidas por cada módulo e é designado por serviço *cron*. A forma mais simples é utilizar um pacote chamado “moodle-cron-for-windows.zip” que torna o procedimento bastante fácil, instalando um pequeno serviço no Windows.

Quanto à autenticação dos alunos, esta vai ser feita por LDAP (*Lightweight Directory Access Protocol*) através do servidor da faculdade. Na fase de testes em que era utilizado o servidor Moodle de laboratório, foram adicionados dois utilizadores, um com privilégios de aluno e outro com privilégios de docente, a fim de se testar o correcto funcionamento da ferramenta de reserva. Uma vez concluída a adaptação da ferramenta, esta vai ser transferida para o Moodle da FEUP que suporta a autenticação *LDAP*. Assim sendo os alunos necessitam unicamente do seu número de aluno e palavra-chave do SIFEUP; assim que introduzidos é acrescentada uma conta de utilizador no Moodle com os dados provenientes do servidor de *LDAP* da faculdade.

5.2 – Mecanismos de reserva de acesso

No que respeita a ferramentas de reserva, foram procuradas mais do que uma solução, ou seja, módulos possíveis de implementar no servidor Moodle e que sejam frequentemente utilizados como blocos de reserva. O módulo usado no protótipo de partida era o WebCalendar, que para cumprir os requisitos pretendidos, requereu alterações ao código fonte. Para além do WebCalendar, foram também encontrados o MRBS Block e o Scheduler Module, que são dois módulos ou blocos de agenda/reserva.

O Scheduler Module é um módulo de actividade que possibilita ao aluno marcar encontros com o professor. O professor define previamente a duração dos blocos de atendimento e o aluno limita-se a fazer a reserva da hora desejada. Após uma análise mais pormenorizada deste bloco, concluímos que não é uma boa opção para o nosso objectivo, pois seria necessária uma vasta alteração do código fonte, para além de este módulo apresentar uma deficiência ao nível do interface. Era pretendida uma visualização deste recurso em forma de calendário semanal dividido em períodos de uma hora, sendo desta forma prático e intuitivo o processo de reserva por parte dos alunos.

O MRBS block é um bloco de actividade desenvolvido para um sistema de reserva de salas de encontros possível de integrar em um servidor Moodle. Este bloco apresenta o interface desejado de calendário dividido em períodos horários. Este bloco não vai ao encontro a todas as necessidades impostas, nomeadamente o lançamento de um hiperligação para o acesso à página da experiência, bem como a vista referente ao calendário, que teria que ser semanal. O código base do MRBS é utilizado para a reserva de mais do que uma sala, ou seja, teria que ser alterado para a reserva de uma única sala (que no nosso caso é uma bancada on-line), ou então no caso de uma possível expansão para múltiplos laboratórios, em que as várias salas corresponderiam às várias bancadas. A grande vantagem do Moodle e seus módulos é serem *software* de código aberto, sendo portanto possível a alteração do código fonte dos módulos, de modo a irem ao encontro às necessidades do utilizador.

O WebCalendar é uma aplicação de calendário baseada em PHP que pode ser configurada para um ou vários utilizadores, para um grupo de utilizadores, ou então, como um calendário de eventos visível pelos visitantes. O calendário pode ser apresentado por dia, semana, mês ou ano, e permite ainda adicionar, editar e remover, tanto utilizador como eventos. Outra funcionalidade útil desta aplicação é a opção por parte do administrador da disciplina de receber notificações via e-mail aquando da inserção, edição e eliminação de eventos por parte dos alunos. Permite também o uso de calendário livre/ocupado, ou seja, só permite adicionar eventos se o calendário estiver livre nesse período. O MySQL, PostgreSQL, Oracle, DB2, Interbase, MS SQL Server, ou ODBC são necessários para o funcionamento desta aplicação. Pode ser configurada para uma variedade de utilizações, como por exemplo:

- Sistema de gestão de horário para uma só pessoa
- Sistema de gestão de horário para um grupo de pessoas, permitindo a um ou mais assistentes supervisionar o calendário de outros utilizadores.
- Um calendário de eventos visível por todos os usuários, permitindo aos visitantes introduzir novos eventos.

Apesar de ser uma aplicação de extrema utilidade, a versão base do WebCalendar não permite a integração directa no Moodle, sendo necessário fazer alterações ao código. Outra inconveniência do uso desta aplicação é a necessidade de serem feitas alterações ao código fonte, sempre que haja uma actualização na versão do Moodle.

5.2.1 – Selecção

A ferramenta de reserva escolhida foi o bloco MRBS, uma vez que quando comparado com os demais, apresenta algumas vantagens importantes, nomeadamente o interface tipo calendário, que pode ser ajustado para visualização diária, semanal ou mensal. Outra vantagem que justifica esta escolha, é o facto de este bloco permitir a reserva e acesso a múltiplas bancadas, ou seja, possibilita ter mais do que uma disciplina, em que a cada uma podem estar associadas mais do que uma experiência. Esta vantagem possibilita a hipótese já pensada, sobre a cooperação entre a FEUP e outras faculdades, podendo os alunos de qualquer uma das instituições aceder à mesma experiência, localizada em diferentes laboratórios. A principal vantagem é que este bloco é desenvolvido inicialmente para ser integrado no Moodle, com o objectivo de permitir a reserva de salas de computadores. Isto implica que são necessárias algumas modificações a nível do código deste bloco, de modo a adaptá-lo ao nosso fim.

As modificações no MRBS reflectem-se ao nível do interface que vai ser apresentado, pois no nosso caso, desejamos reservar o acesso a bancadas e não computadores, e ao nível de

algumas funções do bloco. São exemplo dessas alterações, comparar a data da reserva com a data actual e efectuar o lançamento dos VI's. Outro importante proveito na utilização deste bloco é que sempre que é feita a actualização da versão do servidor de *e-learning* da FEUP, não é necessária nenhuma alteração no código do MRBS, o que não sucedia com a ferramenta de reserva utilizada no protótipo existente. A figura 5.1 ilustra a vista semanal apresentada pelo bloco MRBS. De salientar que nesta figura os períodos já estão divididos em vinte e quatro blocos de uma hora cada, quando em boa verdade a versão original não apresenta esta interface.

Microprocessadores - dado electronico							
<< Go To Week Before		Go To This Week				Go To Week After>>	
Period:	Mon 16 Jun	Tue 17 Jun	Wed 18 Jun	Thu 19 Jun	Fri 20 Jun	Sat 21 Jun	Sun 22 Jun
00h00 - 01h00	*	*	*	*	*	*	*
01h00 - 02h00	*	*	*	*	*	*	*
02h00 - 03h00	*	*	*	*	*	*	*
03h00 - 04h00	*	*	*	*	*	*	*
04h00 - 05h00	*	*	*	*	*	*	*
05h00 - 06h00	*	*	*	*	*	*	*
06h00 - 07h00	*	*	*	*	*	*	*
07h00 - 08h00	*	*	*	*	*	*	*
08h00 - 09h00	*	*	*	*	*	*	*
09h00 - 10h00	*	*	*	*	*	*	*
10h00 - 11h00	*	*	*	*	*	*	*
11h00 - 12h00	*	*	*	*	*	*	*
12h00 - 13h00	*	*	*	*	*	*	*
13h00 - 14h00	*	*	*	*	*	*	*
14h00 - 15h00	*	*	*	*	*	*	*
15h00 - 16h00	*	*	*	*	*	*	*
16h00 - 17h00	*	*	*	*	*	*	*
17h00 - 18h00	*	*	*	*	*	*	*
18h00 - 19h00	*	*	*	*	*	*	*
19h00 - 20h00	*	*	*	*	*	*	*
20h00 - 21h00	*	*	*	*	*	*	*
21h00 - 22h00	*	*	*	*	*	*	*
22h00 - 23h00	*	*	*	*	*	*	*
23h00 - 00h00	*	*	*	*	*	*	*

Figura 5.1 - Vista semanal da ferramenta de reserva (MRBS)

5.2.2 – Integração no Moodle

O MRBS, tal como já foi referido, é um bloco desenvolvido para integrar no Moodle. Apesar de ainda não fazer parte da sua versão base, é extremamente fácil de instalar. O primeiro passo é fazer o download do ficheiro comprimido do bloco *MRBS*, que pode ser feito na página do Moodle, no endereço “<http://www.moodle.org/mod/forum/discuss.php?d=38604#p408505>”. Após efectuado o download do ficheiro, é necessário descompactá-lo e mover os ficheiros para a pasta “*blocks*”, que se encontra na directoria em que foi instalado o Moodle. Uma vez concluído este passo, é necessário aceder à página do servidor Moodle e efectuar o login como administrador, de modo a ter acesso à barra de ferramentas de administração do servidor. Nessa barra de ferramentas, que se encontra na página inicial do servidor Moodle, encontra-se a hiperligação “notificações”, que permite ao servidor verificar a instalação de alguma aplicação. Se a cópia da pasta foi correctamente efectuada, deverá visualizar na página do Moodle a informação sobre a adição das novas tabelas na base de dados e a confirmação que a operação foi efectuada com sucesso. O próximo passo é aceder à configuração deste bloco através do caminho “Notifications > Modules > Blocks > Resource Scheduling” e acrescentar no campo do endereço URL o caminho onde se encontra instalado o *MRBS*, isto é: “<http://.../moodle/blocks/mrbs/web>”. Desta forma, o bloco MRBS já se encontra correctamente instalado na sua versão original.

Para além da descrição do processo de instalação do bloco, é também imprescindível a descrição de todas as alterações efectuadas, de modo a ajustar o bloco aos objectivos

pretendidos. As primeiras alterações e também as mais básicas, são a alteração do nome do bloco e do nome da hiperligação de acesso ao MRBS, que serão visualizados no Moodle. Estas duas alterações devem ser efectuadas no ficheiro “block_mrbs.PHP”, que se encontra no directório “C:\...\moodle\blocks\mrbs\lang\en_utf8 “. Estas mudanças em nada alteram o funcionamento do bloco, mas sim o seu nome, que na sua definição inicial é “*Resource Scheduling*”, e o nome da hiperligação, que é “Schedule a Resource (*Computer Room*) ”.

As seguintes alterações, fundamentais para esta ferramenta ir de encontro dos requisitos propostos, são efectuadas no ficheiro “config.inc.PHP”, que se encontra no directório “C:\...\moodle\blocks\mrbs\web”. A definição inicial do bloco apresentava no quadro de reservas uma interface temporal composta por doze períodos, sendo a desejada de vinte e quatro períodos de uma hora, correspondentes às 24 horas diárias. É assim necessário acrescentar mais doze períodos e mudar o seu formato de apresentação, de períodos de um a doze, para o formato de intervalos de tempo de uma hora, respectivamente diferenciados. A definição inicial do *MRBS* apresentava uma vista diária, sendo a pretendida a semanal. O dia predefinido em que começa a semana é alterado de domingo para segunda-feira. É possível também activar a definição de notificar o administrador e o professor, sempre que é adicionada uma nova reserva. Devido à sua extensão, encontram-se nos anexos as tabelas com o código e as linhas respectivas, onde foi editado código, com o fim de adaptar o bloco *MRBS* aos requisitos desejados. Nas tabelas 9.1 e 9.2 encontram-se enumeradas as alterações necessárias para implementar as funções acima descritas, através da comparação dos campos mudados entre o ficheiro original e o final.

Quando o utilizador pretende efectuar uma reserva, são apresentados campos desnecessários na página de confirmar reserva, que fazem parte da versão base do *MRBS*. Desses campos, são exemplos a possibilidade de repetição diária, semanal ou mensal da reserva. Dessa forma foi necessário proceder à remoção desses campos, sem que isso afectasse o correcto funcionamento do bloco, tarefa que se torna delicada, uma vez que estamos a lidar com reservas em que os dados são gravados em bases de dados. Na tabela 9.3 encontram-se descritas as remoções de código efectuadas no ficheiro “edit_entry.PHP”, que permite eliminar os campos desnecessários.

Outro objectivo do bloco *MRBS* é disponibilizar a hiperligação de acesso à experiência. A melhor solução é fazer o lançamento desta hiperligação na página de visualização da reserva, onde se encontram os dados sobre quem a efectuou, incluindo o dia, a hora e a experiência à qual se pretende aceder. Nesta página encontram-se ainda as opções de edição da reserva e possível remoção, que apenas são disponíveis se o utilizador for o mesmo que a efectuou. A introdução da hiperligação que permite o acesso ao laboratório remoto é feita no ficheiro “view_entry.PHP” que se encontra no directório “C:\...\moodle\blocks\mrbs\web” através da adição do código, presente na tabela 9.4. Este código para além de implementar o lançamento da hiperligação, só a torna visível, se o utilizador que acede à página da reserva for o mesmo que a efectuou.

Foram criados dois ficheiros, “comunicacao.PHP” e “controlo.PHP”, que servem unicamente para abrir as duas páginas que permitem o acesso à experiência remota, dividida em frames, impossibilitando a directa visualização do endereço da página. Apesar de esta medida não ser de elevada segurança, é de fácil implementação, e esconde o endereço da página, tal como pretendido. O código que implementa o lançamento da página Web de comunicação e a do controlo, encontra-se na tabela 9.7 e na tabela 9.8, respectivamente.

Uma vez efectuadas as alterações que adequam esta ferramenta aos nossos requisitos, é necessário por fim atribuir aos utilizadores as permissões de utilização desta ferramenta. Esta atribuição é feita no servidor de *e-learning* por um dos seus administradores. Após feita a autenticação como administrador, acede-se ao menu lateral seguindo o caminho “Users > Permissions > Define roles”, onde depois é feita a atribuição de permissões aos criadores de disciplinas, professores, professores não editores, alunos, convidados, etc.

Uma vez feitas todas as alterações acima descritas, o bloco MRBS, nossa ferramenta de reserva, está pronto a ser utilizado e vai ao encontro dos requisitos impostos.

5.3 – Instalação do *software* e criação dos VI's

Como já foi referido em capítulos anteriores o *software* utilizado para a criação dos VI's foi o LabView, da National Instruments, pelo que o computador utilizado como servidor de laboratório tem que ter uma versão deste *software* instalada. Um dos requisitos deste trabalho é a possibilidade de visualização em tempo real da bancada ou de alguma parte do circuito em particular. Para tal é necessária a criação de um VI que permita a publicação da imagem adquirida pela câmara Web. O primeiro problema com que nos deparámos foi o facto de o *software* LabView, por si só, não suportar a publicação da imagem de uma câmara através de um VI. Após procura na internet de modo a solucionar este problema, foi concluído que era necessária uma aplicação que permitisse a aquisição directa de imagens para o LabView. A solução encontrada é a aplicação “NI-IMAQ for USB Cameras”, que permite a aquisição de imagens de câmaras, câmaras Web, scanners e microscópios, que quando combinada com o *software* “NI Vision Development Module”, permite a aquisição de imagem para um VI. Uma vez instalado o *software* necessário para desenvolver os VI's pretendidos, o próximo passo foi passar à sua criação.

Tal como já descrito no quarto capítulo, de modo a cobrir a totalidade de experiências a realizar na cadeira de Microprocessadores, é necessária a criação de um ou vários VI's de modo a termos as possibilidades de comunicação com a carta, de interacção, de visualização e de instrumentação.

O VI ideal deveria integrar no painel frontal a interacção com a carta, a aquisição imagem da câmara Web, um osciloscópio e um gerador de sinais simplificado, o que permitia a publicação de unicamente dois VI's, o referido e o de comunicação. A criação de tal VI, para além de cobrir a totalidade de experiências a realizar na cadeira de Microprocessadores, também evita o lançamento de demasiados painéis de controlo. Em conjunto com este é necessário efectuar também o lançamento do VI de comunicação com a carta. Após iniciada a criação do referido VI, deparámo-nos com a impossibilidade de conjugar todas estas ferramentas em um único painel, pois a velocidade processamento não é suficiente para responder com eficiência a este conjunto de blocos. Para além disso, acresciam-se ainda problemas de sincronismo temporal entre os vários blocos, uma vez que a comunicação com o DAQ era partilhada por mais do que um bloco dentro desse VI. Assim sendo, tivemos que dividir o VI ideal em dois VI's, o de interacção e visualização e o de instrumentação.

O VI que implementa a interacção com a carta CORE51 já estava previamente implementado, bastando por isso adicionar a possibilidade de aquisição de imagem da câmara Web instalada no laboratório. Desta forma foi necessário editar este VI e adicionar na janela do bloco de diagramas o módulo “USB Acquisition”, e posteriormente configurar este bloco

correctamente com o dispositivo de entrada e formato da imagem a ser apresentada no painel frontal. O painel frontal respectivo a este VI está ilustrado na figura 5.2.

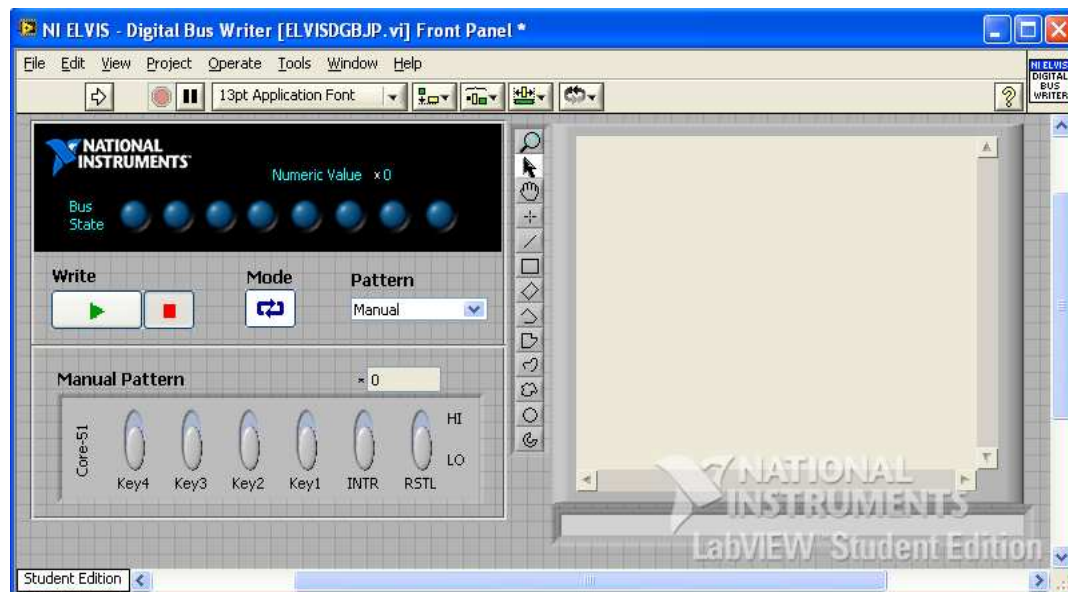


Figura 5.2 - VI de interação com a CORE51

De forma a desenvolver o painel de instrumentação, era necessário integrar em um único VI um osciloscópio e um gerador de sinais simplificado. Ambos os VI's se encontravam disponíveis através do *software* de instalação da NI ELVIS, o que à primeira vista pode induzir-nos no erro de considerar tarefa fácil a sua junção. Este processo não é de todo trivial, pois a programação em LabView não permite a simples opção de copiar o VI original e colá-lo em outra janela. Desta forma foi necessário escolher qual dos VI's iria ser tomado como inicial e que seria editado a fim de implementar o restante. Como o VI que apresenta maior grau de complexidade é o osciloscópio, este foi editado a fim de lhe ser acrescentado um gerador de funções simplificado. Uma vez que estamos a editar um VI *standard*, incluído no pacote NI ELVIS, convém fazer uma cópia para uma directoria diferente, de modo a não alterar o VI original. Na janela do bloco de diagramas foi editado o VI do osciloscópio, sendo acrescentados os ciclos e blocos que, aliados aos apropriados controladores e indicadores, permitiram implementar o gerador de funções. A figura 5.3 apresenta o resultado final da integração destes dois VI's em um único painel frontal.

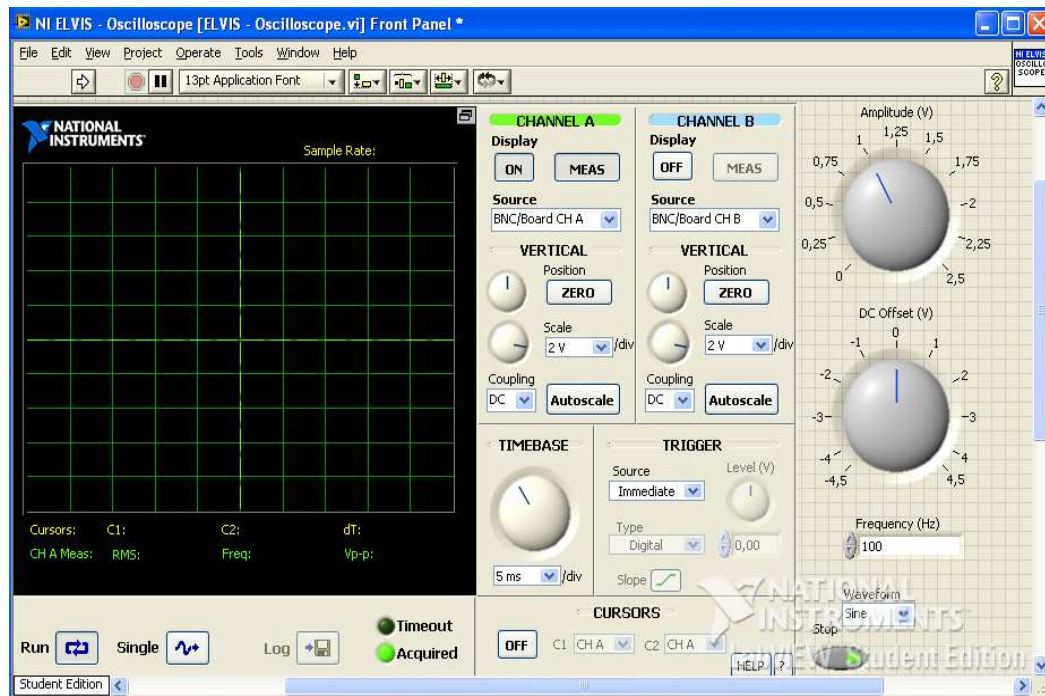


Figura 5.3 - VI de instrumentação

Este VI, tal como desejado, apresenta as seguintes funcionalidades:

- Osciloscópio:
 1. Visualização de dois sinais em simultâneo ou alternados
 2. Regulação da escala vertical, ou seja, da grandeza a medir
 3. Regulação da escala horizontal, ou seja, da escala temporal
 4. *Trigger* com qualquer um dos dois canais existentes
- Gerador de funções:
 5. Aplicação de uma forma de onda com amplitude a variar entre 0 e 2,5 V
 6. *Offset CC* a variar entre -4,5 a 4,5 V
 7. Frequência variável de 0 a 250 kHz
 8. Aplicação de uma onda sinusoidal, quadrada ou triangular.

Por fim resta o painel de comunicação com a carta *CORE51*, que foi o único que não houve a necessidade de criar nem alterar, pois o protótipo existente já apresentava este VI a funcionar da forma desejada. O painel frontal de comunicação está representado na figura 5.4.

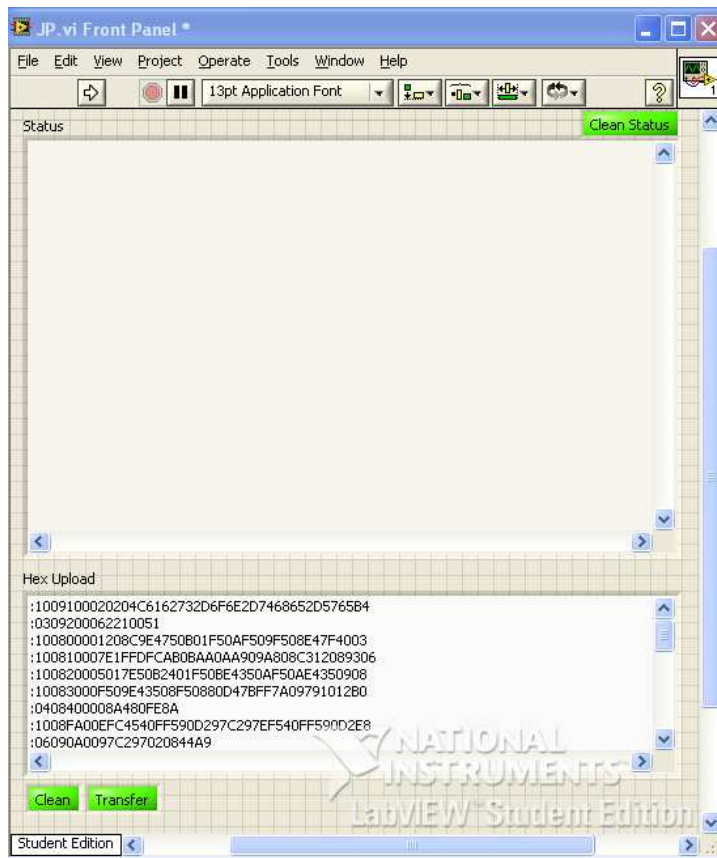


Figura 5.4 - VI de comunicação

De salientar que a programação em *LabView* é gráfica, pelo que se torna impraticável descrever todos os passos para a criação dos VI's acima descritos. Desta forma encontram-se como anexo em formato digital, todos os ficheiros relativos a estes painéis, permitindo assim a visualização da programação utilizada.

Apresentados os VI's criados, é necessário ainda referir a ligação existente entre o painel de comunicação e o painel de controlo, que em certa parte funcionam em sincronismo. Como já foi explicado anteriormente, é através do VI de controlo que é escolhido o modo de operação da carta CORE51, através da sua selecção no menu *dropdown*, assim como é através do VI de comunicação que é feita transferência de dados com a CORE51. Assim sendo, a comunicação com a CORE51 vai ser influenciada pelo modo de operação escolhido, o que obriga a que haja uma comunicação entre estes dois VI's para que o de comunicação tenha conhecimento de qual o modo escolhido pelo utilizador. A cada modo de operação possível de escolher no menu *dropdown*, está associado um valor numérico que vai ser interpretado pelo VI de comunicação. No bloco de diagramas deste VI existe uma estrutura de eventos, que faz corresponder o respectivo evento a cada valor do modo de operação escolhido.

No caso da selecção do *cold test* é activado sequencialmente cada um dos sinais de controlo e é retornado para a janela de estados do VI de comunicação o estado da memória da CORE51.

Quanto ao modo *hex upload*, assim que executado, o painel de comunicação recebe na caixa de texto *Status* a informação sobre o estado da memória e indica ao utilizador que se encontra à espera da transferência do código. Uma vez colado o código na caixa de texto *Hex*

upload e pressionado o botão de *Transfer*, é iniciada a transferência do código para a CORE51. Assim que ela terminar, a CORE51 fica à espera do carácter ASCII de espaço, de forma a dar início ao programa, informação essa que também é apresentada em *Status*.

No caso da selecção do modo manual no painel de controlo, o VI de comunicação acciona o evento respectivo a este modo, que corresponde à aplicação dos sinais de controlo (*array* de interruptores) nas teclas da CORE51.

5.4 – *Hardware* da bancada on-line

Apesar de a maioria do *hardware* utilizado na implementação desta bancada on-line já ter sido referido no decorrer dos capítulos anteriores, é necessário descrever com mais pormenor todo o *hardware* utilizado.

De forma evidente pode-se considerar o computador, que vai ser o servidor de laboratório e o principal componente de *hardware* que constitui a bancada, uma vez que os restantes componentes serão obrigatoriamente a ele ligados. O computador necessita também de ter interface para ligação RS232, pois a comunicação com a CORE51 é efectuada através deste *standard*. Assim sendo, a CORE51 é outro componente de *hardware* a utilizar nesta bancada e talvez um dos mais significativos, pois em boa verdade é onde se encontra o poder de processamento da bancada de trabalho, ou seja, da experiência a realizar. Para além desta interface, o computador necessita ainda de uma placa para se ligar à rede da FEUP e colocar a experiência disponível via Web.

Outra parte importante desta bancada é a plataforma NI ELVIS, que veio revolucionar o acesso remoto a experiências e à qual é ainda ligada a carta CORE51. Visto que a interface *standard* de um computador não permite a sua ligação directa com a NI ELVIS, é necessária a utilização de uma placa de aquisição (NI-DAQ), que vem em conjunto com a plataforma. É necessária também a placa que permite a criação de protótipos e a ligação com a NI ELVIS, que desta forma possibilita o acesso a todas as entradas, saídas e instrumentação da plataforma, através dos conectores disponibilizados.

Tal como já foi referido em outros capítulos, existe a necessidade de uma imagem em tempo real, por isso uma câmara USB faz também parte do *hardware* que integra a bancada on-line.

Para além do *hardware* já mencionado, pode haver a necessidade de utilizar outros componentes, que se justificam pela natureza da experiência a realizar. São exemplos, variados tipos de componentes electrónicos que vão ser utilizados na área de protótipo.

5.4.1 – A carta CORE51 e a ligação à estação NI ELVIS

A placa CORE51 ilustrada na figura 5.3 destina-se ao desenvolvimento de programas para a família 51 da Intel. Permite uma melhor familiarização com os modos de funcionamento dos periféricos internos dos microcontroladores dessa família, bem como de muitos outros compatíveis com ela. Pode ser utilizada como um sistema autónomo para auto-aprendizagem ou como núcleo de circuitos mais complexos, dispondo para isso de todos os sinais necessários à sua expansão.

O desenvolvimento de programas para esta placa deve ser feito tendo em conta que serão executados a partir do endereço 0x0000. Podem ser desenvolvidos em *assembly* ou qualquer linguagem de alto nível para a qual exista compilador.

Para interface com o utilizador dispõe de quatro *LED*'s, activos a zero, ligados de P1.4 (*LED* 1) a P1.7 (*LED* 4), e de quatro teclas, activas a zero, ligadas de P1.3 (*TECLA* 1) a P1.0 (*TECLA* 4). Dispõe ainda de uma tecla (*RESET*) para inicialização manual do programa e outra (*INTR*) para geração de uma interrupção.

Para proceder à ligação e teste da placa, esta deve ser ligada à saída de 12V da fonte de alimentação e a uma porta série do computador e correr uma aplicação de emulação de terminal (*Teraterm* ou *Hyperterminal*) que permitem, em caso de necessidade, acompanhar o auto teste da placa através das mensagens que ela envia. Para isso devem pressionar-se todas as seis teclas da placa, largando-se de seguida largar a tecla *RESET* e só depois as restantes. Se tudo estiver funcional os *LED*'s da placa acendem alternadamente e no ecrã do PC aparece uma mensagem de confirmação.

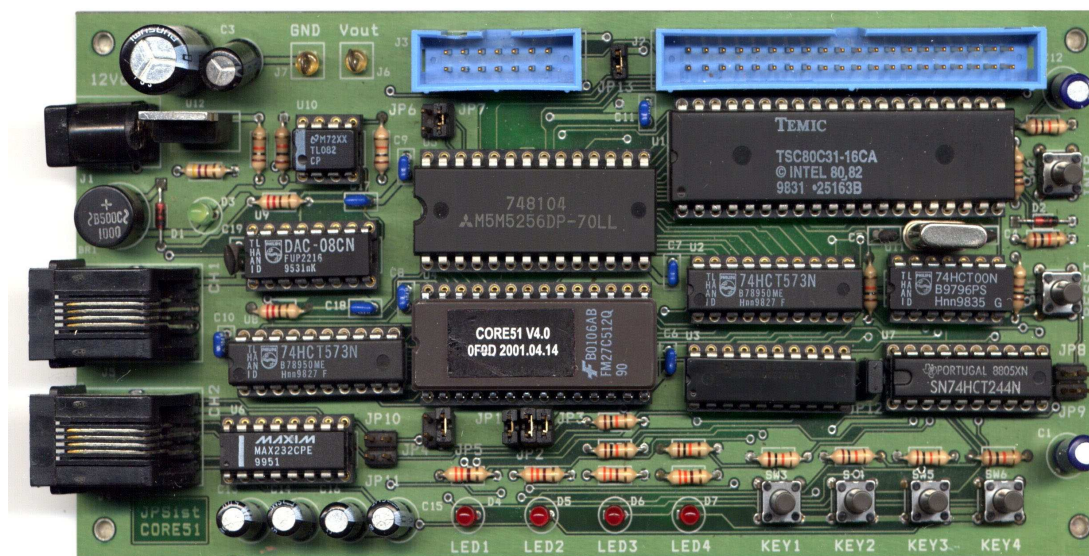


Figura 5.5 - Carta CORE51 [origem: 8]

A integração da CORE51 com a NI ELVIS é obtida através do VI de comunicação e do de interacção com a carta. Em boa verdade a CORE51 e a plataforma NI ELVIS não se encontram fisicamente ligados por meio de algum interface, mas sim através de *software* que comunica com ambas. A CORE51 apenas faz parte de um circuito que está montado na placa de protótipo da NI ELVIS.

5.4.2 – Adaptação a outros Microcontroladores/Microprocessadores

Esta bancada tem como objectivo o ensino da cadeira de Microprocessadores através do uso do microcontrolador da família 80C51, mas essa limitação pode ser ultrapassada com algumas modificações. Independentemente do microprocessador ou microcontrolador a utilizar, a lista de VI's necessários não se alteraria, bem como o *hardware* acima descrito, com a excepção da carta CORE51. Esta como sabemos permite uma maior familiarização com os

modos de funcionamento dos periféricos internos dos microcontroladores da família 80C51, mas tal como estes, outros cumpririam com a mesma ou até maior eficiência os requisitos impostos para esta bancada.

Uma alternativa que foi também estudada durante o desenvolvimento deste trabalho foi a carta Keil MCB900, que só não foi utilizada devido à actual familiarização com a CORE51, bem como a já existência do VI de comunicação na versão protótipo. Independentemente da posterior opção pela CORE51, a carta Keil MCB 900, ilustrada na figura 5.6, foi objecto de estudo detalhado, pelo que apesar da sua não inclusão nesta bancada, pode vir a ser incluída em bancadas futuras. Desta forma é de boa prática apresentar um sumário das suas características:



Figura 5.6 - Placa Keil MCB900 [origem: 9]

- Microcontrolador P89LP932/5 da *Philips* montado num PLCC *socket*
- XTAL a 7,3 MHz
- Possui ISD51 interface (nova tecnologia de monitorização para programas do 8051)
- Memória RAM integrada de 0,7K
- Memória Flash integrada de 8K
- 8 LEDs ligados a E/S de microcontrolador
- Entrada analógica (potenciómetro)
- Porta série (RS232)
- Área de protótipo para o circuito do utilizador
- Dimensões da placa (58 x 110 mm)
- Tensão de alimentação de 5-9 VCC
- Corrente típica de 50 mA
- Corrente máxima de 100 mA

Esta carta apresenta três modos de funcionamento distintos, sendo dois deles fundamentais para a realização de uma experiência. Apesar do modo *uVision2/ISD51* ser pouco utilizado, ou até mesmo dispensável para o nosso projecto, os dois restantes modos, *Flash Magic* e *User Run*, são indispensáveis. A escolha do modo de funcionamento é feita manualmente através de três *jumper*s que se encontram na placa, ou seja, inserindo umas cápsulas que permitem a sua activação. Uma vez que a experiência deve ser acedida à distância, torna-se primordial encontrar uma solução que permita efectuar esta selecção

remotamente. Tal será possível através da integração de três micro-relés na área de protótipo da carta, sendo o controlo feito através de três entradas digitais acessíveis ao utilizador. Desta forma, torna-se fundamental o uso do VI referente ao *array* de interruptores que vai permitir ao utilizador actuar nos relés através das entradas digitais, permitindo seleccionar o modo de funcionamento desejado. Esse VI já foi apresentado no capítulo referente à biblioteca de VI's necessária para abranger a totalidade de experiências a serem desenvolvidas na cadeira de Microprocessadores.

Uma vez resolvido o problema da selecção dos vários modos de funcionamento, é importante descrever os vários passos a efectuar de modo a realizar uma experiência remota com esta carta. O primeiro modo é o *Flash Magic*, que permite carregar o código para a memória do microcontrolador. De salientar que neste modo é efectuado o *reset* da carta, pois um dos *jumpers* a ser activado é o do *Reset*. Uma vez carregado o código na memória *flash*, é seleccionado o modo *User Run*, que permite ao utilizador executar o programa armazenado na memória do microcontrolador. Na tabela 5.1 apresentamos os vários modos de funcionamento associados à selecção dos *jumpers*.

Tabela 5.1 - Modos de operação da Keil MCB900 [origem: 10]

<i>Jumpers</i>	Modo de Operação		
	Flash Magic	User Run	µVision/ISD51
Run (3.3v fixos em V_{DD})	OFF	ON	OFF
Reset (via porta COM)	ON	OFF	ON

Quanto ao *jumper* AV, deve ser usado para conectar o potenciómetro ao pino de entrada AD12, que é um conversor A/D.

Através dos *jumpers*, a placa MCB900 pode ser configurada para funcionar nos seguintes modos [8]:

- *FlashMagic* – programando a *Flash* ROM do P89LCP932, que usa um carregador *ISP Flash* especial
- *uVision2/ISD51* – *In-System Debugging* com a *Keil ISD51*
- *User Run* – executa a aplicação guardada na *Flash ROM* da P89LPC932

De modo a ser possível a integração desta carta na bancada é necessário efectuar algumas alterações nos VI's de comunicação e de interacção. A CORE51 recebe o código Intel HEX por comunicação RS232C, e o mesmo sucede com a Keil MCB900. No caso da CORE51, assim que efectuada a transferência do código, esta fica à espera do carácter ASCII de espaço para confirmar a execução, sendo a escolha do modo de funcionamento feita através do VI de interacção com a carta. Já no caso da Keil MCB900 a selecção do modo de funcionamento é feita através dos *jumpers*, que teriam que ser orientados através de micro-relés, que por sua vez eram controlados pelo VI de interacção com a carta através das entradas digitais existentes. Uma vez que o VI de comunicação e o de interacção se encontram irredutivelmente associados, teriam que ser feitas algumas modificações na programação de ambos.

Capítulo 6

Apresentação da bancada

Ao contrário do capítulo anterior, que apresentou uma abordagem maioritariamente técnica, que visava descrever todo o processo de realização da bancada on-line, bem como servir de guia prático de apoio à sua utilização, o presente capítulo tem como objectivo a apresentação final da bancada. Vão ser descritos pormenorizadamente todos os passos que devem ser executados pelo utilizador para realizar uma experiência, desde a autenticação e reserva no Moodle, até ao interface final que permite ao utilizador controlar as experiências. É também apresentado um exemplo prático de todo o processo de realização de uma experiência através do acesso on-line a uma bancada situada em um laboratório do DEEC.

6.1 – Autenticação e reserva da bancada

Tal como já foi referido em capítulos anteriores, pretendeu-se que a ferramenta de reserva estivesse disponível no servidor de *e-learning* da FEUP, ou seja, através do endereço “<http://moodle.fe.up.pt>”. Uma vez que não foi ainda efectuada a integração do bloco MRBS no servidor Moodle da FEUP por ser necessária uma série de testes que apenas se realizam em períodos de transição, foi disponibilizado o servidor Moodle de teste do próximo ano lectivo, que é usado antes de ser efectuada alguma alteração no servidor principal da faculdade. Desta forma foi-nos atribuída uma conta com privilégios de professor, para que pudéssemos adicionar disciplinas e experiências ao bloco MRBS. Em boa verdade, o facto de ser utilizado o servidor de testes, em nada prejudicou o objectivo pretendido, pois este servidor também suportava a autenticação via LDAP, que é uma das finalidades de integração deste bloco no servidor da FEUP. Desta forma foi utilizado o servidor Moodle de teste que é acessível a partir do endereço “<http://moodle.fe.up.pt/dev0809>”, ao invés do acima mencionado.

Assim que este bloco seja submetido aos respectivos testes e posteriormente adicionado no Moodle de produção, os utilizadores poderão aceder à bancada on-line através do servidor Moodle da faculdade.

O utilizador para, poder realizar uma experiência on-line, deve aceder à página <http://moodle.fe.up.pt/dev0809> e efectuar o login com os dados do SIFEUP.

Após efectuado o login, vai ser visualizada a página principal do Moodle, onde se encontra no canto inferior direito, a hiperligação de acesso à ferramenta de reserva, com o nome de “*Labs On The Web*”, onde o utilizador deve clicar.

Uma vez na página do bloco MRBS, é visualizado um calendário semanal, dividido em blocos de uma hora cada, em que o utilizador pode ver o actual estado de reservas da bancada e, se o desejar, efectuar a sua própria reserva. Na figura 6.1 é apresentado o interface visualizado pelo utilizador quando pretende adicionar uma reserva.

vhftorres moodle
Labs On The Web

24 Jun 2008 goto Help Admin Report Search: You are vhftorres Log Off

Add Entry

Brief Description: vhftorres

Date: 24 Jun 2008

Period: 07h00 - 08h00

Experiences: dado electronico

Save

View Day: 18 Jun | 19 Jun | 20 Jun | 21 Jun | 22 Jun | 23 Jun | 24 Jun | 25 Jun | 26 Jun | 27 Jun | 28 Jun | 29 Jun | 30 Jun | 01 Jul
View Week: 26 May | 02 Jun | 09 Jun | 16 Jun | 23 Jun | 30 Jun | 07 Jul | 14 Jul | 21 Jul
View Month: Apr 2008 | May 2008 | Jun 2008 | Jul 2008 | Aug 2008 | Sep 2008 | Oct 2008 | Nov 2008 | Dec 2008

Print Preview

Figura 6.1 - Confirmação da reserva

É permitido ao utilizador aceder à página de reserva de outros utilizadores, a fim de saber quem efectuou a reserva, mas as opções de editar ou eliminar essa reserva só estão disponíveis para quem a efectuou e para o professor da cadeira. A figura 6.2 apresenta o aspecto de uma página de reserva.

vhftorres moodle
Labs On The Web

24 Jun 2008 goto Help Admin Report Search: You are vhftorres Log Off

vhftorres

Experience: Microprocessadores - dado electronico
Schedule: 07h00 - 08h00, Tuesday 24 June 2008
Created By: vhftorres

Edit Entry
Delete Entry
Return to previous page

View Day: 18 Jun | 19 Jun | 20 Jun | 21 Jun | 22 Jun | 23 Jun | 24 Jun | 25 Jun | 26 Jun | 27 Jun | 28 Jun | 29 Jun | 30 Jun | 01 Jul
View Week: 26 May | 02 Jun | 09 Jun | 16 Jun | 23 Jun | 30 Jun | 07 Jul | 14 Jul | 21 Jul
View Month: Apr 2008 | May 2008 | Jun 2008 | Jul 2008 | Aug 2008 | Sep 2008 | Oct 2008 | Nov 2008 | Dec 2008

Print Preview

Figura 6.2 - Visualização da reserva

Apesar de nesta fase de teste apenas se encontrar uma bancada operacional, no futuro podem ser implementadas mais do que uma, podendo o aluno escolher qual a bancada a que deseja aceder. Assim sendo, existe um menu *dropdown* (figura 6.3) que permite escolher uma de entre as possíveis experiências, sendo carregado o calendário de reservas referente à bancada seleccionada.



Figura 6.3 - Selecção da experiência no caso de multi-laboratório

Em cada um dos blocos de uma hora ainda não reservados encontra-se um sinal de adição que, uma vez clicado, permite ao utilizador efectuar a reserva para a hora escolhida. Uma vez feita a reserva, o sinal de adição previamente existente no período de acesso pretendido é substituído por uma hiperligação com o nome do utilizador. Esta hiperligação permite a todos os utilizadores aceder à página da respectiva reserva, sendo atribuídos unicamente privilégios de edição ao utilizador que a fez. Deste modo, o utilizador pode editar ou até mesmo eliminar a reserva, se assim o pretender, bastando para tal aceder à página respectiva e efectuar as acções pretendidas. Assim que chega a hora de acesso reservada, é lançada uma hiperligação que permite ao aluno aceder aos VI's da experiência on-line.

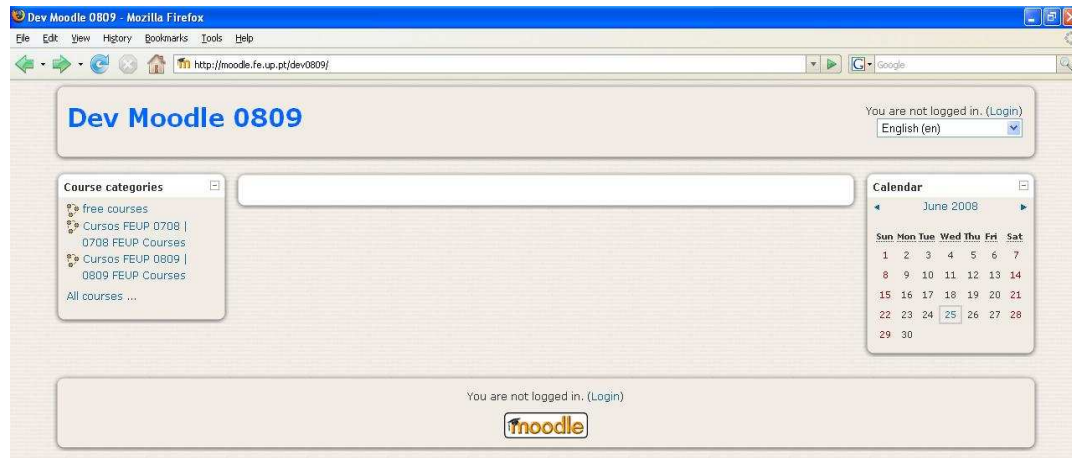


Figura 6.4 - Página inicial do Moodle da FEUP na sua versão de testes

6.2 – Interface com a bancada

A interacção entre o utilizador e a bancada é efectuada através de duas páginas Web, que são lançadas pelo bloco MRBS. Uma das páginas Web apresenta-se dividida em dois *frames*, nos quais vão ser carregados o VI de interacção com a carta e o VI de instrumentação. Na outra página Web é carregado o VI de comunicação. Uma vez que já foi descrito o processo de criação de cada VI, essa abordagem não vai ser repetida, sendo apenas explicado o modo de controlo dos painéis através das páginas Web carregadas.

Começando pelo VI de comunicação, o utilizador pode limpar as duas caixas de texto existentes, a superior e a inferior. A superior não é editável e apenas serve como interface de

informação sobre o estado da comunicação com a CORE51 e o seu modo de funcionamento. Quanto à caixa inferior é onde deve ser colado o código em Intel HEX, que será transferido para a carta através do botão com esse nome. Na caixa superior será visualizada informação relativa à comunicação e à transferência. Se tudo estiver a funcionar correctamente é pedido ao utilizador para pressionar a tecla de espaço, a fim de executar o programa transferido, tal como mostra a figura 6.5.

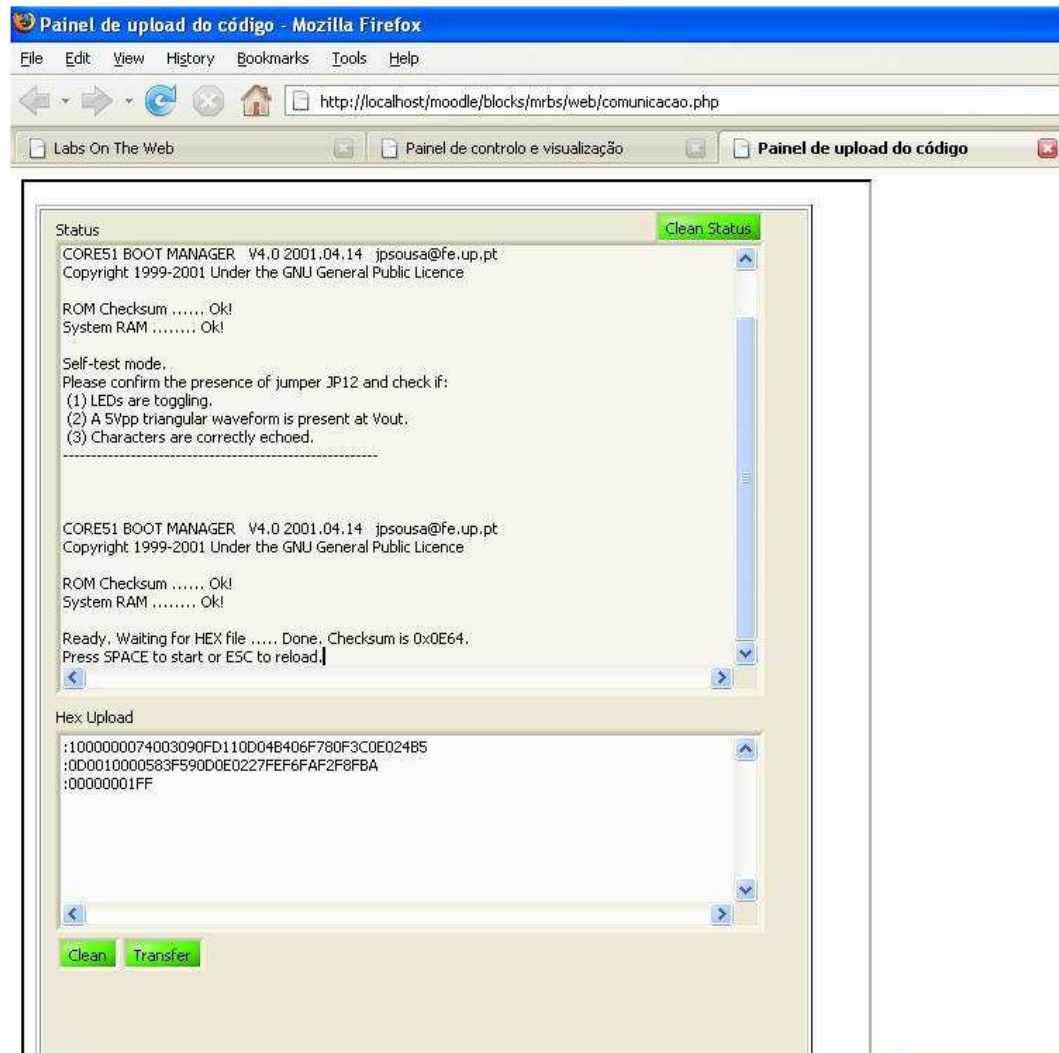


Figura 6.5 - Janela Web do VI de comunicação

Os restantes dois VI's serão visualizados em duas *frames* e permitem a posterior visualização e controlo da experiência. A figura 6.6 apresenta na *frame* superior o VI de interacção com a CORE51, ficando assim o VI de instrumentação na *frame* inferior.

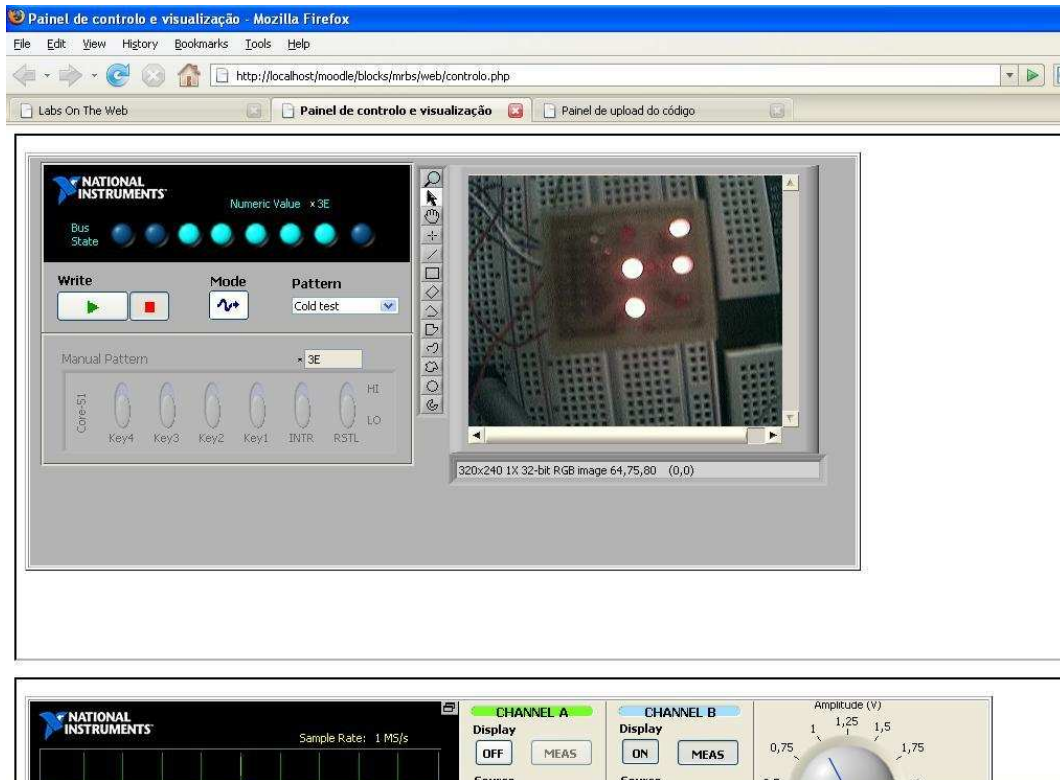


Figura 6.6 - Janela Web do VI de interacção com a experiência

Quanto ao VI de interacção, este apresenta um menu *dropdown* que permite a escolha do modo de funcionamento da CORE51. Assim que seleccionado o modo desejado é necessário clicar o botão *play* para dar inicio à execução do modo escolhido. Existe ainda o botão *stop* para interromper a execução do modo *manual*. Uma vez que o modo *manual* é o que permite o controlo da experiência por parte do utilizador, é necessário ainda descrever a função do *array* de seis botões existentes neste VI. Tal como já foi referido no capítulo anterior, a CORE51 apresenta um botão de interrupção, um de *reset* e quatro ligados às teclas de entrada da carta, que são controlados pelo *array* de seis botões. Ainda nesta *frame* é possível de visualizar uma imagem em tempo real da experiência, adquirida através de uma câmara Web.

O VI de instrumentação não apresenta nenhuma complexidade quanto à sua utilização, visto que se trata de um interface muito prático, que relembra os instrumentos reais que os alunos já estão certamente habituados a usar nas aulas práticas. O osciloscópio permite a visualização de um ou de dois canais em simultâneo, bem como a regulação das escalas vertical e horizontal. Para além disso, possibilita ainda o *trigger* com qualquer um dos canais e o acoplamento CC ou CA. Quanto ao gerador de funções, permite a selecção da amplitude, frequência e *offset* do sinal a aplicar, bem como a escolha do tipo de onda, através de um menu *dropdown*. O gerador de funções encontra-se previamente desligado, pelo que se for pretendida a sua utilização é necessário clicar no botão de activação respectivo.

6.3 – Exemplo de aplicação

De modo a que o utilizador tenha uma clara e correcta ideia do acesso on-line a uma experiência da cadeira de Microprocessadores, é exemplificado neste subcapítulo todo o

procedimento de reserva e acesso à bancada que implementa o lançamento de um dado electrónico.

O primeiro passo é aceder à página do Moodle, através do endereço “http://moodle.fe.up.pt/dev0809”. De seguida o utilizador deve efectuar o *login* com os seus dados do SIFEUP. Uma vez registado, deve ser visualizado na página principal do Moodle a hiperligação com o nome de “Labs On The Web”, que permite o acesso ao laboratório on-line, tal como exemplificado na figura 6.7.

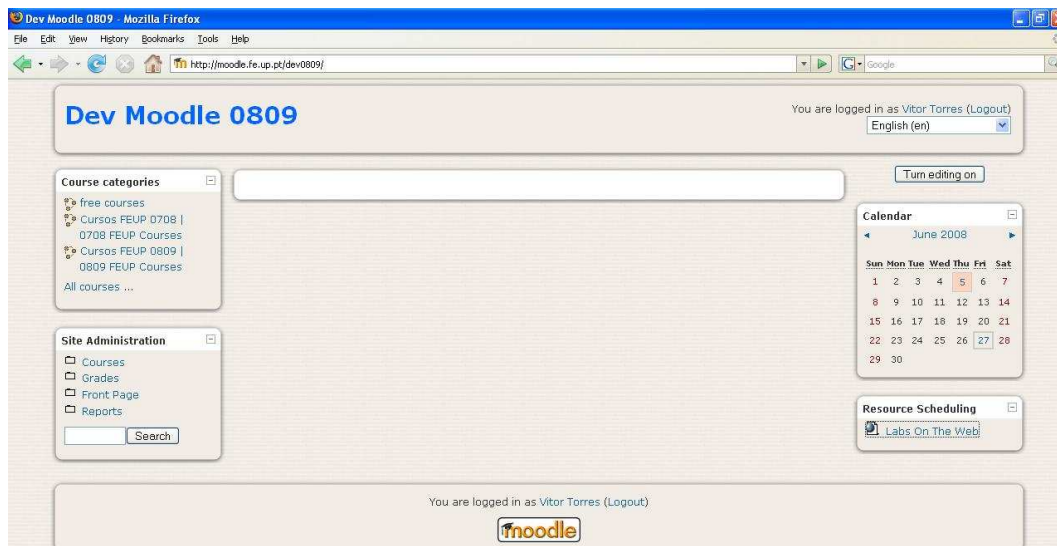


Figura 6.7 - Página inicial do Moodle de testes

Após clicarmos nessa hiperligação, é-nos apresentado um calendário semanal, a começar na corrente semana, com um menu *dropdown* que permite escolher a experiência a realizar. No caso da figura 6.3 são indicadas de três experiências com o intuito de dar a entender a possibilidade de múltiplo laboratório. Neste caso o menu *dropdown* apresenta apenas a experiência do dado electrónico, pois é a única que se encontra disponível.

Após efectuar a reserva e chegado o respectivo intervalo, acedemos à página correspondente clicando no nosso nome de utilizador, que aparece no período respectivo, de modo a podermos aceder à experiência. Se o acesso a esta página for dentro do tempo por nós reservado, aparece uma hiperligação que permite o acesso à experiência on-line, tal como mostra a figura 6.8.

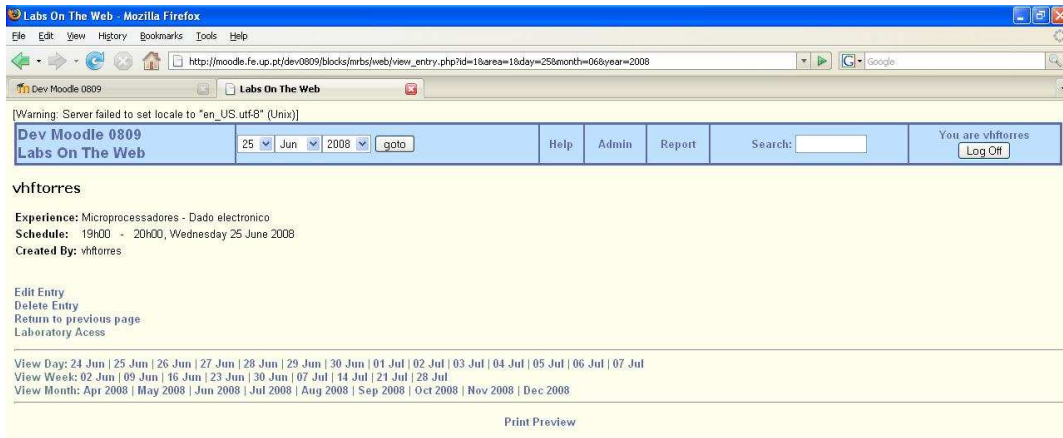


Figura 6.8 - Página da reserva

Assim que clicarmos nessa hiperligação, são lançadas duas janelas de navegação, onde serão carregados os dois VI's que permitem a completa interacção com a experiência, tal como se pode verificar na figura 6.9.

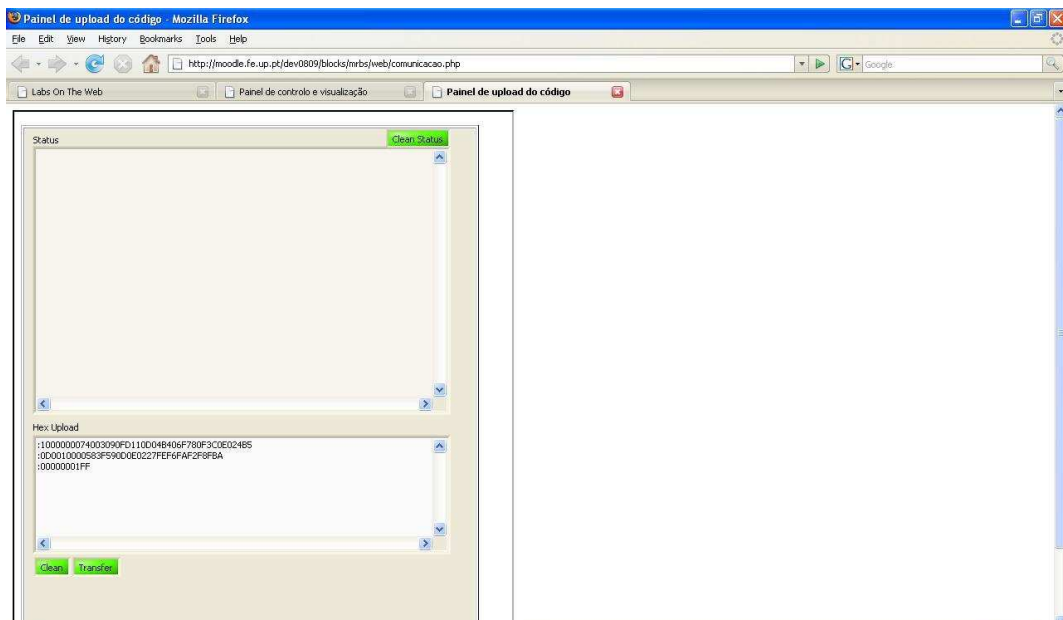


Figura 6.9 - Janelas de navegação com os dois VI's carregados

O próximo passo é seleccionar e executar o modo *cold test*, de forma a verificar o estado da CORE51 e da sua comunicação. Se a informação obtida na caixa de texto *Status* for positiva, procedemos à transferência do código hexadecimal. Para tal executamos o modo *hex upload* e de seguida colamos o código objecto do programa na caixa de texto respectiva. Assim que carregamos no botão *transfer* é iniciada a transferência do código e é dada a informação de que a CORE51 se encontra à espera do carácter “espaço” para executar o programa. Esta informação pode ser comprovada através do exemplo da figura 6.10.

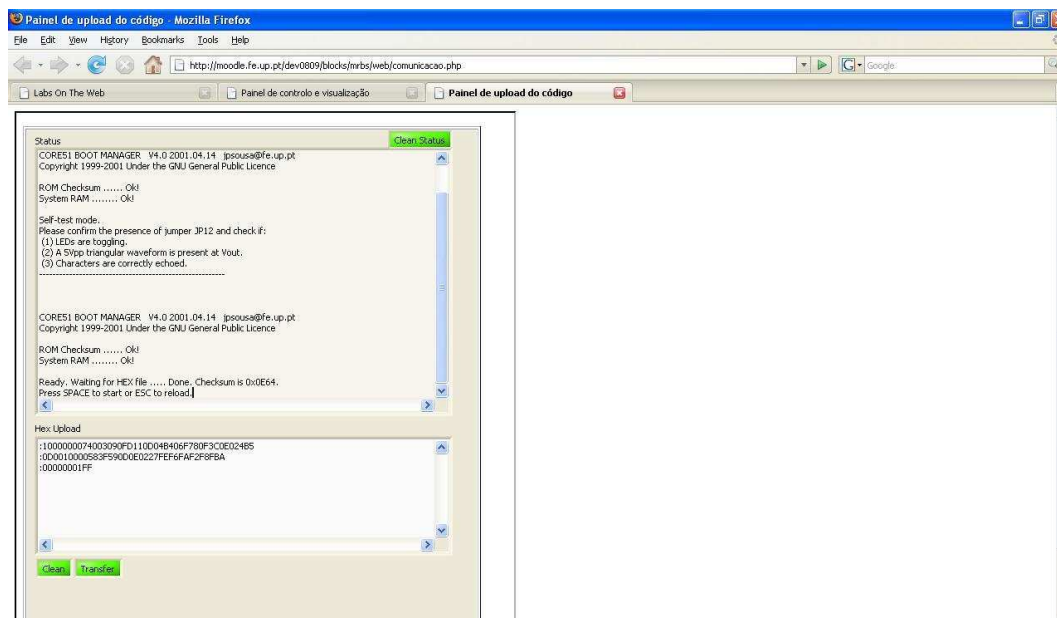


Figura 6.10 - Transferência do código hexadecimal

Por último, é colocado em execução o modo manual, que é o que nos permite interagir com a experiência. Uma vez que no caso do lançamento de um dado electrónico não é necessária a utilização do osciloscópio, foram ligadas aos dois canais deste, as saídas que permitem acender os LED's correspondentes ao resultado um e ao resultado dois. Uma vez que os LED's acendem sempre que a saída do microcontrolador a eles ligada estiver a nível lógico "zero", devemos visualizar no osciloscópio um sinal de amplitude zero no caso de sair algum dos números referidos. As figuras 6.11 e 6.12 exibem os casos em que saíram os resultados, dois e três, respectivamente. De salientar que no caso do número três, encontram-se acesos os LED's correspondentes ao acender do resultado um e do dois, daí serem observados dois sinais a nível lógico zero. Já no caso do resultado dois, tal com verificado, obtemos um sinal a nível lógico "um" e outro a "zero".

Uma vez efectuados todos os lançamentos desejados, fechamos o navegador Web de forma a pormos termo à experiência.

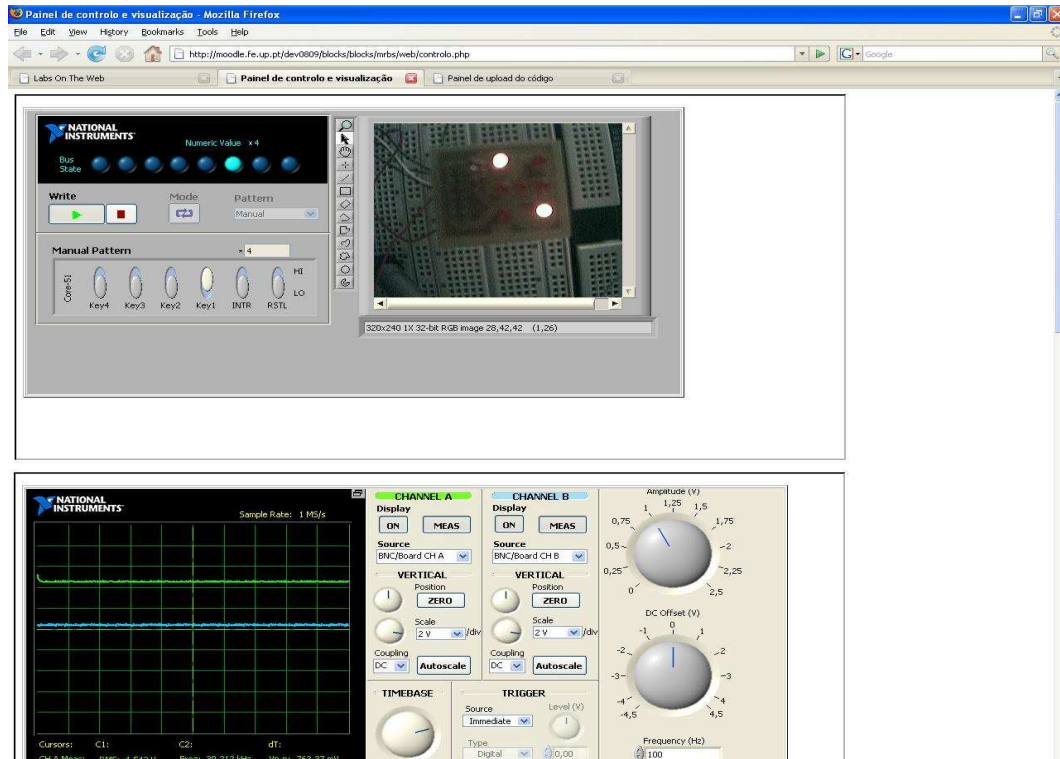


Figura 6.11 - Lançamento do dado em que saiu o número dois

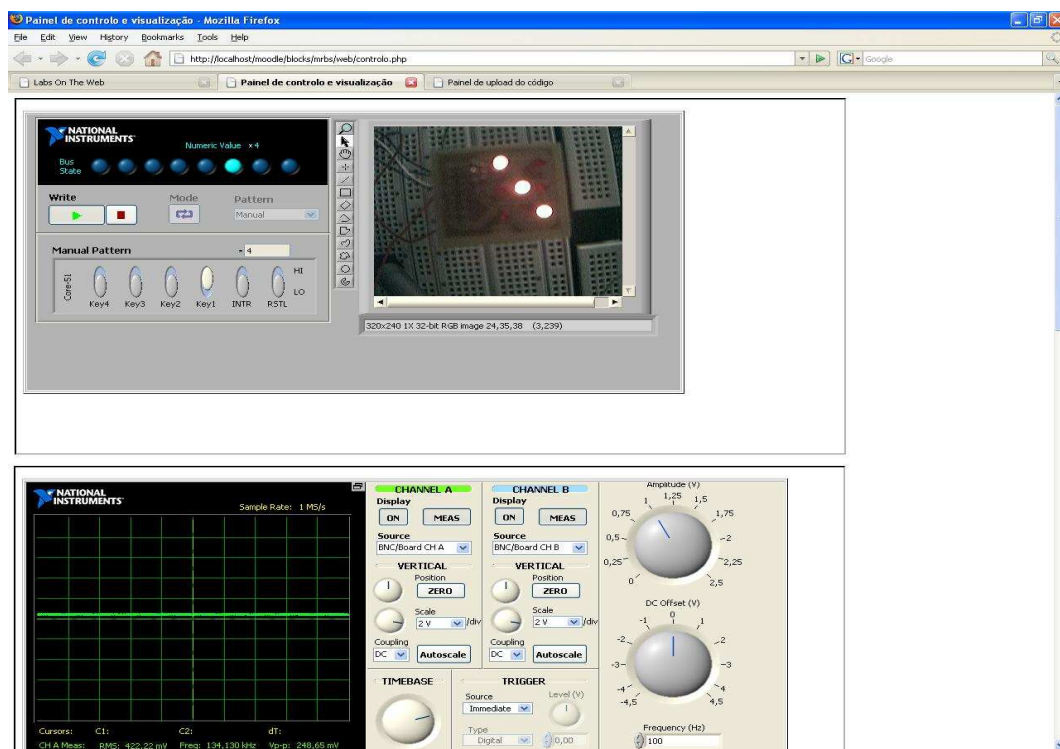


Figura 6.12 - Lançamento do dado em que saiu o número três

Capítulo 7

Conclusão

Este capítulo efectua a avaliação final do trabalho desenvolvido e indica possíveis direcções de futuro. Esta análise é feita relativamente aos objectivos alcançados, ao grau de sucesso com que foram implementados e também aos que não foram alcançados, mas que podem ser alvos de desenvolvimento em trabalhos futuros.

7.1 – Análise crítica

No início deste documento foi descrito o objectivo do trabalho, que resumidamente era implementar uma bancada on-line para apoiar o ensino da cadeira de Microprocessadores. Em boa verdade posso afirmar que esse objectivo foi atingido, se bem que como em quase todo o projecto e devido à escassez de tempo, houve certas partes que ficaram por aperfeiçoar. Quanto aos VI's de acesso à bancada on-line, foram desenvolvidos de forma a serem o mais funcionais possíveis e a abranger um amplo leque de experiências com interesse na cadeira de Microprocessadores, objectivo esse que foi concluído com êxito. Já no que respeita ao mecanismo de reserva, o MRBS é uma ferramenta de grande utilidade e que, devido à sua natureza de código aberto permite uma fácil adaptação às nossas necessidades. Devido à escassez de tempo, não foi possível desenvolver esta bancada em cooperação com uma universidade do Brasil, a fim de esses alunos poderem aceder à bancada localizada no DEEC (e aos alunos da FEUP poderem aceder a uma bancada situada no Brasil). Outro ponto que não foi possível de melhorar é a segurança no acesso à bancada, pois apesar de os VI's serem lançados em *frames*, que ocultam automaticamente o endereço URL, é relativamente fácil obtê-lo. De salientar que apesar desta imperfeição, apenas os alunos da faculdade têm acesso à bancada, visto que é preciso estar ligado localmente à rede da FEUP, no recinto da faculdade ou no exterior, através de uma ligação VPN.

Resumindo, pode afirmar-se que os objectivos principais deste trabalho foram alcançados dentro das limitações iniciais referentes à familiarização com esta tecnologia e também às linguagens de programação usadas.

7.2 – Direcções de futuro

Este subcapítulo aparece evidentemente associado ao anterior, uma vez que algumas das possíveis direcções de futuro resultam de alguns objectivos que não foram alcançados, ou que merecem uma optimização.

A primeira sugestão como direcção de futuro é a parceria com uma faculdade do Brasil, tal como já referido na análise crítica, permitindo também aos alunos das várias faculdades a realização de um maior conjunto de experiências, bem como a troca de informação e conhecimento.

Em segundo lugar, a optimização do sistema de lançamento do acesso à experiência é também um importante aspecto a melhorar, respectivamente no que diz respeito à segurança. O protótipo existente apresentava um URL de acesso aos VI's em que era constituído por uma parte hexadecimal, renovada de hora em hora. Devido à falta de documentação não nos foi possível implementá-la dentro dos limites de tempo impostos. Assim sendo, a aplicação deste mecanismo na bancada, ou de outro semelhante pode ser alvo de desenvolvimento futuro.

Por último, a utilização de vídeo-conferência não foi integrada nos VI's, uma vez que os tornaria ainda mais pesados, para além de que o *software* Macromedia Flash Player utilizado na versão de protótipo, não é um *software* grátis, o que obriga à compra de licenças e posteriores renovações. Para além disso, a actual existência de *softwares* livres que permitem vídeo-conferência, como é o caso do Skype e do MSN tornam menos útil a integração destes recursos no VI.

Referências

- [1] M. Teresa Restivo, Fernando G Almeida, M. Fátima Chouzal, Joaquim Mendes e António M. Lopes. Laboratórios Remotos: monitorização e actuação via Web. Faculdade de Engenharia da Universidade do Porto, Dezembro de 2006.
- [2] J. M. Martins Ferreira, Elisabete Almeida, Isabel Figueiredo, Carlinda Leite. Labs-on-the-Web – A multidisciplinary project to evaluate the pedagogical effectiveness of on-line Labs. EDEN 2008, Lisboa, 11-14 Junho.
- [3] Ensino e pesquisa, National Instruments, disponível em <http://digital.ni.com/worldwide/brazil.nsf/web/all/EDE6AB35934211EC8625733F006EC139>. Acedido em 20/Maio/2008
- [4] Lamar University, disponível em <http://www.ee.lamar.edu/EELABS/images/NI-ELVIS.jpg>. Acedido em 14/Maio/2008.
- [5] Ni traz um Conjunto para Projecto e Construção de Protótipos baseados em LabView para estudantes de Engenharia e Ciências, National Instruments, disponível em <http://digital.ni.com/worldwide/brazil.nsf/web/all>
- [6] Alberto José Ávares, João Carlos Espíndola Ferreira. Metodologia para implantação de laboratórios remotos via internet na área de automação da manufactura. ABCM 2003, 18-21 de Maio.
- [7] Moodle da FEUP, instalação do Moodle, disponível em <http://moodle.fe.up.pt/2005/doc/?file=install.html>. Acedido em 12/Março/2008.
- [8] João Paulo Filipe de Sousa. Familiarização com a CORE51. Faculdade de Engenharia da Universidade do Porto, Outubro de 2005.
- [9] Equinox Techonolies, disponível em http://www.equinox-tech.com/products/images/keil/mcb900_230.jpg. Acedido em 13/Junho/2008.
- [10] Keil, Embedded Development Tools, disponível em http://www.keil.com/support/man/docs/mcb900/mcb900_overview.htm. Acedido em 13/Junho/2008.

Tabela 9.3 - Remoção dos campos desnecessários

Código a remover	Linhas
<pre> <TR><TD CLASS=TR> <?php echo get_vocab("fulldescription");?> </TD> <TD CLASS=TL> <TEXTAREA NAME="description" ROWS=8 COLS=40 WRAP="virtual"> <?php echo htmlspecialchars (\$description); ?> </TEXTAREA> </TD> </TR> </pre>	296 a 298
<pre> <?php } ?> <TR><TD CLASS=CR><?php echo get_vocab("duration");?></TD> <TD CLASS=CL><INPUT NAME="duration" SIZE=7 VALUE="<?php echo \$duration;?>"> <SELECT NAME="dur_units"> <?php if(\$enable_periods) \$units = array("periods", "days"); else \$units = array("minutes", "hours", "days", "weeks"); while (list(,\$unit) = each(\$units)) { echo "<OPTION VALUE=\$unit"; if (\$dur_units == get_vocab(\$unit)) echo " SELECTED"; echo ">".get_vocab(\$unit); } ?> </SELECT> <INPUT NAME="all_day" TYPE="checkbox" VALUE="yes" onClick="OnAllDayClick(this)"> <?php echo get_vocab("all_day"); ?> </TD></TR> </pre>	332 a 355
<pre> <TR><TD CLASS=CR><?php echo get_vocab("type");?></TD> <TD CLASS=CL><SELECT NAME="type"> </pre>	459 a 467

<pre><?php for (\$c = "A"; \$c <= "J"; \$c++) { if (!empty(\$typeI[\$c])) echo "<OPTION VALUE=\$c" . (\$type == \$c ? " SELECTED" : "") . ">\$typeI[\$c]\n"; } ?></SELECT></TD></TR></pre>	
<pre><TR> <TD CLASS=CR><?php echo get_vocab("rep_type")?></TD> <TD CLASS=CL></pre>	472 a 473
<pre>echo "<INPUT NAME=\"rep_type\" TYPE=\"RADIO\" VALUE=\"\" . \$i . \"\""; if(\$i == \$rep_type) echo " CHECKED"; echo ">" . get_vocab("rep_type_\$i") . "\n";</pre>	478 a 483
<pre><TR> <TD CLASS=CR><?php echo get_vocab("rep_end_date")?></TD> <TD CLASS=CL><?php genDateSelector("rep_end_", \$rep_end_day, \$rep_end_month, \$rep_end_year) ?></TD> </TR> <TR> <TD CLASS=CR><?php echo get_vocab("rep_rep_day")?> <?php echo get_vocab("rep_for_weekly")?></TD></pre>	490 a 497
<pre><?php for (\$i = 0; \$i < 7; \$i++) { \$wday = (\$i + \$weekstarts) % 7; echo "<INPUT NAME=\"rep_day[\$wday]\" TYPE=CHECKBOX"; if (\$rep_day[\$wday]) echo " CHECKED";</pre>	500 a 506

<pre> echo ">" . day_name(\$wday) . "\n"; } </pre>	
<pre> <TR> <TD CLASS=CR><?php echo get_vocab("rep_num_weeks")?> <?php echo get_vocab("rep_for_nweekly")?></TD> <TD CLASS=CL><INPUT TYPE=TEXT NAME="rep_num_weeks" VALUE="<?php echo \$rep_num_weeks?>"> </TR> </pre>	547 a 550

Tabela 9.4 - Lançamento da hiperligação de acesso aos painéis de controlo

Código a acrescentar	Linhas
<pre> <script language="Javascript"> //lançamento das dois painéis em duas janelas separadas function teste() { window.open ("controlo.php"); window.open ("comunicacao.php"); } </script> <?php if (\$year == \$yr) { if(\$month == \$mth2) { if(\$day == \$dy) { if(\$start_date[0] == \$info[0]) { if(\$start_date[1] == \$info[1]) { if(\$user == \$row[0]) { //lançamento do link com chamada da função java que lança as duas janelas assim que efectuado click ?> Laboratory Acess <?php } } } } } } </pre>	353 a 385

<pre> } } } ?> </pre>	
--------------------------------------	--

Tabela 9.5 - Compatibilização dos dados temporais

Código a acrescentar	Linhas
<pre> <?php \$user = getUsername(); #utilizador que esta a visualizar a reserva \$id = required_param('id', PARAM_INT); #utilizador que efectuou a reserva mrbsGetEntryInfo(\$id); #obter as informações da reserva que retorna o array de strings row[0] \$time = time(); \$info = userdate(\$time, '%H'); #da a hora actual \$yr = date("Y"); \$mth = date("M"); \$dy = date("d"); if (\$dy[0] == 0) //se o primeiro numero for um 0, elimina o 0 a fim de comparar com o dia da reserva pois no formato em que se encontra elimina automaticamente o 0 { \$dy = \$dy[1]; } date("d Y "); //transformacao do formato do mes, de abreviatura para numeros a fim de compatibilizar o mes actual e o da reserva </pre>	279 a 350

```
if($mth == Jan)
{
    $mth2 = 1;
}
if($mth == Feb)
{
    $mth2 = 2;
}
if($mth == Mar)
{
    $mth2 = 3;
}
if($mth == Apr)
{
    $mth2 = 4;
}
if($mth == May)
{
    $mth2 = 5;
}
if($mth == Jun)
{
    $mth2 = 6;
}
if($mth == Jul)
{
    $mth2 = 7;
}
```


<pre> if(\$mth == Aug) { \$mth2 = 8; } if(\$mth == Sep) { \$mth2 = 9; } if(\$mth == Oct) { \$mth2 = 10; } if(\$mth == Nov) { \$mth2 = 11; } if(\$mth == Dec) { \$mth2 = 12; } ?> </pre>	
--	--

Tabela 9.6 - Endereços de acesso aos VI's

Código a acrescentar
<pre><?php \$link_cont = "http://apocalypse.pdc-ptse.fe.up.pt:85/ELVISDGBJP.html"; \$link_comu = "http://apocalypse.pdc-ptse.fe.up.pt:85/JP.html"; \$link_medi = "http://apocalypse.pdc-ptse.fe.up.pt:85/ELVIS%20-%20Oscilloscope.html"; ?></pre>

Tabela 9.7 - Ficheiro comunicacao.PHP

Código a acrescentar
<pre><html> <head> <meta http-equiv="Content-Type" content="text/html; charset=windows-1252"> <title>Painel de upload do código</title> </head> <body> <?php include "links.php"; ?> <p><iframe name="I1" width="600" height="700" src="<?php echo \$link_comu?>"> </iframe></P> </body> </html></pre>

Tabela 9.8 - Ficheiro controlo.PHP

Código a acrescentar
<pre><html> <head> <meta http-equiv="Content-Type" content="text/html; charset=windows-1252"> <title>Painel de controlo e visualização</title> </head> <body> <?php include "links.php"; ?> <p><iframe name="I1" width="1000" height="500" src="<?php echo \$link_cont?>"> </iframe></P> <iframe name="I2" width="1000" height="620" src="<?php echo \$link_medi?>"> </iframe></p> </body> </html></pre>

Anexo B: Enunciado do trabalho EEC0029 relativo ao dado electrónico

Apresentação

O trabalho proposto neste documento tem por objectivo principal reforçar as competências dos alunos no domínio da análise e síntese de código *assembly* para a família 51. O exemplo escolhido para atingir estes objectivos consiste na realização de um “dado electrónico”, usando-se para este efeito a porta 1 do microcontrolador: as linhas P1.0 a P1.3 são usadas para ler quatro teclas, sendo as restantes linhas desta porta (P1.4 a P1.7) usadas para comandar sete leds, da forma que se representa abaixo.



Repare-se que a saída P1.7 é a única que alimenta apenas um led (o do meio), já que as restantes alimentam sempre um par de leds (cada par acende / apaga ao mesmo tempo). Sempre que uma saída se encontrar em 0, o(s) led(s) a ela ligado(s) acende(m).

Tarefas propostas

Comece por preencher a tabela apresentada abaixo, para saber o que deve escrever-se em P1.7 a P1.4 (e em P1.3 a P1.0?), de modo a visualizar os seis resultados possíveis.



Recorra ao ambiente de desenvolvimento da KEIL (µVision 3) para realizar as seguintes tarefas:

1. Abrir o ficheiro assembly “dado.a51” (disponível na página da disciplina juntamente com este guião), que contém o código apresentado adiante. Identifique as razões que impedem este código de produzir os resultados esperados e efectue as alterações necessárias para corrigir os erros encontrados.

2. Depois de introduzir as correcções pedidas, acrescente o código necessário para: **a)** controlar o “Start / Stop” através da entrada de interrupção /INT0; **b)** suportar uma entrada de batota (na linha P1.1) que triplique a probabilidade do resultado “4”.

LOC	OBJ	LINE	SOURCE
		1	
----		2	cseg at 0
		3	
0000		4	inicio:
0000 7400		5	mov a,#0
0002		6	marca1:
0002 2090FD		7	jb p1.0,\$
0005 12000E		8	call leds
0008 04		9	inc a
0009 B406F6		10	cjne a,#06,marca1
000C 80F2		11	jmp inicio
000E		12	leds:
000E 83		13	movc a,@a+pc
000F F590		14	mov p1,a
0011 22		15	ret
0012 7FEF6FAF		16	db 7fh,0efh,6fh,0afh,2fh,8fh
0016 2F8F		17	
		18	end

3. Introduza as modificações necessários para que o funcionamento com ou sem batota passe a ser controlado pelas entradas P1.3 a P1.0, de acordo com a tabela seguinte:

P1.3 ... P1.0	Modo de funcionamento
0	Sem batota
1 ... 6	O resultado correspondente à combinação presente em P1.2...P1.0 sai com probabilidade tripla
7 e 8	Sem batota
9 ... 14	Sai o resultado correspondente à combinação presente em P1.2...P1.0 (probabilidade 1)
15	Sem batota

4. Recorrendo aos temporizadores internos do 80C51 e às interrupções que eles podem gerar, garanta que cada resultado está presente nos leds durante 2,5 ms.

5. Modifique o código para que se possa seleccionar o modo de funcionamento via RS-232C (substituindo P1.3...P1.0 pela correspondente palavra de 4 bits). A rotina de atendimento à porta série deve receber dois códigos ASCII, de acordo com o seguinte protocolo: **b + 0... 7** corresponde às primeiras oito combinações da tabela e **B + 0 ... 7** às últimas oito linhas. Devem ignorar-se as sequências que não correspondam a uma configuração válida.

6. Por fim, seleccione um microcontrolador da família 51 que permita realizar a funcionalidade pretendida, usando o mínimo espaço possível, e apresente o diagrama esquemático do circuito para esta aplicação (aceitando uma alimentação dc de 7 a 12 V).